

(19) 日本国特許序 (J P)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2002-84339

(P2002-84339A)

(43) 公開日 平成14年3月22日(2002.3.22)

(51) Int.Cl.<sup>7</sup> 識別記号  
 H 0 4 L 29/08  
           12/56      2 0 0  
           13/08  
 H 0 4 N 7/24

F I		午後一時(参考)	
H 0 4 L	12/56	2 0 0 Z	5 C 0 5 9
	13/08		5 K 0 3 0
	13/00	3 0 7 C	5 K 0 3 4
H 0 4 N	7/13	Z	

審査請求 未請求 請求項の数15 O.L. (全 25 頁)

(21)出願番号	特願2001-202147(P2001-202147)
(22)出願日	平成13年7月3日(2001.7.3)
(31)優先権主張番号	特願2000-204632(P2000-204632)
(32)優先日	平成12年7月6日(2000.7.6)
(33)優先権主張国	日本(JP)

(71) 出願人 000005821  
松下電器産業株式会社  
大阪府門真市大字門真1006番地

(72) 発明者 春元 英明  
大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(72) 発明者 堀内 優希  
大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

(74) 代理人 100098291  
弁理士 小笠原 史朗

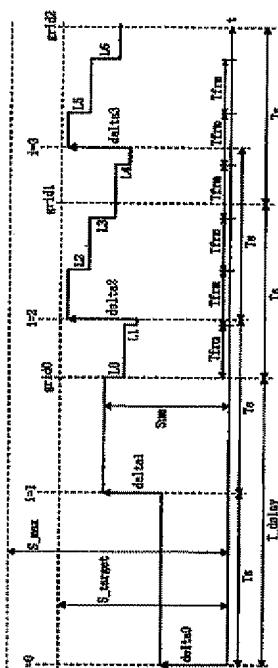
最終頁に続く

(54) 【発明の名称】 ストリーミング方法およびそれを実行するシステム

(57) 【要約】

【課題】 端末のバッファ容量が機種によって異なっていても、ネットワークの伝送能力が変動しても、ストリーミング再生の破綻を回避することが可能であり、しかも、ストリーミング再生の破綻回避と、頭出し時の待ち時間短縮とを互いに両立させることができるようなストリーミング方法を提供する。

**【解決手段】** 端末は、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームの目標量 ( $S_{target}$ ) を決定し、また、バッファ容量を伝送能力で除して得られる値を超えない範囲内で任意に、自身のバッファにストリームの先頭データを書き込んでからそのデータを読み出して再生開始するまでの遅延時間 ( $T_{delay}$ ) を決定して、それら目標量および遅延時間をサーバに通知する。サーバは、通知に基づき、端末のバッファ占有量 ( $Sum$ ) が目標量の近傍で目標量を超えずに遷移するように、送信速度を制御する。



## 【特許請求の範囲】

【請求項1】 サーバが端末へネットワークを通じてストリームデータを送信し、かつ端末が当該ストリームデータを受信しつつ再生するストリーミング方法であって、

端末が、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定ステップ、

当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、端末が、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定ステップ、

決定した目標時間および遅延時間を、端末がサーバに通知するステップ、

サーバが端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御ステップを備える、ストリーミング方法。

【請求項2】 前記制御ステップにおいて、サーバは、端末のバッファに蓄積されているストリームデータの量が、当該目標量の近傍において当該目標量を超えることなく遷移するように、当該送信速度を制御することを特徴とする、請求項1に記載のストリーミング方法。

【請求項3】 前記制御ステップにおいて、サーバは、当該送信速度と、当該遅延時間と、端末がストリームデータをデコードする速度とに基づいて、端末のバッファに蓄積されるストリームデータの量を予測算出することを特徴とする、請求項2に記載のストリーミング方法。

【請求項4】 端末が、ネットワークの伝送能力が所定の閾値を跨いで変化したことを検出する検出ステップ、前記検出ステップでの検出結果に応じて、端末が当該目標量を変更する目標量変更ステップ、および変更後の目標量を、端末がサーバに通知するステップをさらに備え、

前記制御ステップにおいて、サーバは、変更後の目標量の通知を受けると、端末のバッファに蓄積されるストリームデータの量が、当該変更後の目標量の近傍において当該変更後の目標量を超えることなく遷移するように、当該送信速度を制御することを特徴とする、請求項1に記載のストリーミング方法。

【請求項5】 前記検出ステップでネットワークの伝送能力が第1の閾値を跨いで低下したことを検出すると、端末は、前記目標量変更ステップにおいて、当該目標量を増加させる向きに変更し、

前記制御ステップにおいて、サーバは、当該目標量が増加されたのに応じて、当該送信速度を上昇させる向きに制御することを特徴とする、請求項4に記載のストリーミング方法。

【請求項6】 当該第1の閾値は、実現可能な最大の伝

送能力と、ストリームデータの転送ロスが発生し始めるような伝送能力との略中間の値であることを特徴とする、請求項5に記載のストリーミング方法。

【請求項7】 前記検出ステップでネットワークの伝送能力が当該第1の閾値より小さい第2の閾値を跨いで低下したことを検出すると、端末は、前記目標量変更ステップにおいて、当該目標量を減少させる向きに変更し、前記制御ステップにおいて、サーバは、当該目標量が減少方向に変更されたのに応じて、当該送信速度を低下させる向きに制御することを特徴とする、請求項4に記載のストリーミング方法。

【請求項8】 当該第2の閾値は、ストリームデータの転送ロスが発生し始めるような伝送能力と対応する値であることを特徴とする、請求項7に記載のストリーミング方法。

【請求項9】 前記目標量変更ステップにおいて、端末が当該目標量を減少させる向きに変更すると、前記制御ステップにおいて、サーバは、送信しようとするストリームを構成する各フレームの再生時刻を現在時刻と逐次比較して、再生時刻が現在時刻よりも古いフレームの送信をスキップし、それよって当該送信速度を低下方向に制御することを特徴とする、請求項8に記載のストリーミング方法。

【請求項10】 前記目標量変更ステップにおいて、端末が当該目標量を減少させる向きに変更すると、前記制御ステップにおいて、サーバは、送信しようとするストリームを構成する各フレームの重要度を基準値と逐次比較して、

重要度が基準値未満であるフレームについては、全て送信をスキップし、

重要度が基準値以上であるフレームについては、それぞれの再生時刻を現在時刻と逐次比較して、再生時刻が現在時刻よりも古いものだけ送信をスキップし、それよって当該送信速度を低下方向に制御することを特徴とする、請求項8に記載のストリーミング方法。

【請求項11】 ネットワークを通じてストリームデータを送信するサーバと、当該ストリームデータを受信しつつ再生する端末とからなるシステムであって、端末は、

自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定手段、

当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定手段、および決定した目標時間および遅延時間をサーバに通知する手段を備え、

サーバは、端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に

に基づいて送信速度を制御する制御手段を備える、システム。

【請求項12】 ネットワークを通じてストリームデータを送信するサーバと共に用いられ、当該ストリームデータを受信しつつ再生する端末であって、

サーバには、端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御手段が備わり、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定手段、

当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定手段、および決定した目標時間および遅延時間をサーバに通知する手段を備える、端末。

【請求項13】 ストリームデータを受信しつつ再生する端末と共に用いられ、ネットワークを通じて当該ストリームデータを送信するサーバであって、

端末には、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定手段、

当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定手段、および決定した目標時間および遅延時間をサーバに通知する手段が備わり、

端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御手段を備え、前記制御手段は、端末のバッファに蓄積されているストリームデータの量が、当該目標量の近傍において当該目標量を超えることなく遷移するように、当該送信速度を制御することを特徴とする、サーバ。

【請求項14】 サーバが端末へネットワークを通じてストリームデータを送信し、かつ端末が当該ストリームデータを受信しつつ再生するストリーミング方法を記述したプログラムであって、

端末が、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定ステップ、

当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、端末が、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定ステップ、

決定した目標時間および遅延時間を、端末がサーバに通

知するステップ、

サーバが端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御ステップを備えるストリーミング方法を記述した、プログラム。

【請求項15】 サーバが端末へネットワークを通じてストリームデータを送信し、かつ端末が当該ストリームデータを受信しつつ再生するストリーミング方法が記述されたプログラムを記録した記録媒体であって、

端末が、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定ステップ、

当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、端末が、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定ステップ、

決定した目標時間および遅延時間を、端末がサーバに通知するステップ、

サーバが端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御ステップを備えるストリーミング方法が記述されたプログラムを記録した、記録媒体。

#### 【発明の詳細な説明】

##### 【0001】

【発明の属する技術分野】 本発明は、ストリーミング方法に関し、より特定的には、サーバが端末へインターネットを通じてマルチメディアデータを送信し、かつ端末がそのデータを受信しつつ再生するためのストリーミング方法に関する。

##### 【0002】

【従来の技術】 (マルチメディアデータの符号化圧縮方式およびバッファモデルの説明) インターネットでの伝送に使用されるマルチメディアデータには、動画、静止画、音声、テキスト、およびそれらが多重化されたデータ等、さまざまな種類がある。動画では、H.263やMPEG1、2、4といった符号化圧縮方式が著名であるし、静止画としてはJPEG、音声では、MPEGオーディオ、G.729など枚挙にいとまがない。

【0003】 本発明では、ストリーミング再生に目的を絞っているので、動画および音声が伝送の対象となる。ここでは、動画圧縮方式の代表であるMPEGビデオ、中でも比較的仕組みが単純なMPEG1 (ISO/IEC 11172) ビデオや、MPEG2 (ISO/IEC 13818) ビデオについて説明する。

【0004】 MPEGビデオは、高効率なデータ圧縮を実現するために、主に次の2つの特徴を有している。一つ目は、動画像データの圧縮において、従来から行われていた空間周波数特性を用いた圧縮方式の他に、フレー

ム間での時間相関特性を用いた圧縮方式を取り入れたことである。MPEGでは、ストリームを構成している各フレーム（ピクチャとも呼ぶ）を、Iフレーム（フレーム内符号化ピクチャ）、Pフレーム（フレーム内符号化と過去からの参照関係を使用したピクチャ）、Bフレーム（フレーム内符号化と過去および未来からの参照関係を使用したピクチャ）の3種類に分類してデータ圧縮を行う。これらの3種では、Iフレームが最も大きく（つまり情報量が最も多く）、次いでP、Bの順である。圧縮アルゴリズムにも大きく依存するが、情報量の比は、およそ I : P : B = 4 : 2 : 1 程度となる。また一般的に、MPEGビデオストリームは、15フレーム（= 1 GOP）を単位として、1 GOPについて I フレームが1枚、P フレームが4枚、B フレームが10枚の割合で含まれている。

【0005】MPEGビデオの二つ目の特徴は、画像の複雑さに応じた動的な符号量割り当てをピクチャ単位で行える点である。MPEGのデコーダは、デコーダバッファを備え、このデコーダバッファにデータを蓄積してからデコードを行うことで、圧縮の難しい複雑な画像に対して大量の符号量を割り当てることが可能になっている。MPEGに限らず動画圧縮では、標準的なデコーダバッファの容量を規格で定義する場合が殆どである。MPEG1やMPEG2の場合、標準デコーダバッファは、規格で容量が 224 KByte と定義されており、MPEGエンコーダは、この容量の範囲内でデコーダバッファ占有量が遷移するようにピクチャデータを生成しなければならない。

【0006】図19(A)～(C)は、従来のストリーミング方法を説明するための図である。図19(A)は、ビデオフレームを示す図、図19(B)は、バッファ占有量の遷移を模式的に示した図、図19(C)は、従来端末の構成例を示す図である。図19(C)において、端末は、ビデオバッファと、ビデオデコーダと、I、P並べ替えバッファと、スイッチとを備えている。ビデオバッファが、前述のデコーダバッファに相当し、転送されてくるデータは、ビデオバッファに蓄積された後、ビデオデコーダによってデコードされる。デコードされたデータは、I、P並べ替えバッファおよびスイッチを通じて再生時刻順に並べ替えられる。

【0007】図19(B)において、縦軸はバッファ占有量（ビデオバッファのデータ蓄積量）を、横軸は時間を示し、図中の太線がバッファ占有量の時間的遷移を示している。また、太線の傾きは、ビデオのビットレートに相当し、一定のレートでデータがバッファに入力されていることを示している。また、一定間隔（33.3667 msec）でバッファ占有量の削減が起こっているが、これは、一定周期で各フレームのデータがデコードされていくことによる。また、斜め点線と時間軸との交点は、各ビデオフレーム内のデータがビデオバッファへ

向けて転送開始される時刻を示している。従って、図19(A)に示されたフレームXの転送開始時刻は t1、フレームYの転送開始時刻は t2 となる。

【0008】図19(A)、(B)において、ビデオの先頭フレームXがビデオバッファに入力開始される時刻 t1 から、最初にデコードが実行される時刻（図中、太線の第1の立ち下がり位置）までの時間を一般に、vbv\_delay 時間と呼ぶ。最初のデコードは、ビデオバッファが満杯になった瞬間に実行されるので、vbv\_delay 時間は、通常、データ入力開始から容量 224 KByte のビデオバッファが満杯になるまでの時間であり、従って、ビデオの入力が開始されてから、デコーダを通じてビデオ再生が開始されるまでの初期遅延時間（頭出し時の待ち時間）ということになる。

【0009】図19(A)のフレームYが複雑な画像である場合、図19(B)に示されているように、その符号量が大量なので、フレームYのデコード時刻（図中の t3）よりも早い時刻（図中の t2）から、ビデオバッファへのデータ転送を開始しなければならない。ただし、どんなに複雑な画像でも、バッファを占有するピクチャ量は、224 KByte の許容範囲内である。

【0010】図19(B)に示したバッファ遷移がきちんと保たれるようにビデオバッファにデータが転送されるならば、ビデオバッファのアンダーフローやオーバーフローによるストリーミング破綻が起こらないことは、MPEGの規格で保証されている。

【0011】（ネットワーク転送ジッタ吸収用の受信バッファの説明）ところが、図20に示すように、サーバ201と端末202とをネットワーク203で接続し、ストレージ210中のMPEGデータを配信する場合、生成モジュール211でパケットを生成する時間や、ネットワーク機器204、205における転送手続き時間、ネットワーク203の混雑などに伴なう伝送遅延時間などのために、データの転送レートに揺れが生じる。従って、実際には、図19(B)に示したバッファ遷移が保たれないのが実情である。このような転送レートの揺れ（ジッタ）を緩和吸収する方法としては、まず、ネットワークの帯域に比べ十分小さい符号化レートのコンテンツを流すことが考えられる。しかし、ネットワーク資源をできる限り有効に使って高品位な映像や音声を提供する必要があるので、この方法は適切ではない。そこで、一般には、ネットワーク機器204、205に、それぞれ適当な容量の送受信バッファ206、207を設け、普段からデータを多少先送り気味に転送しておいて、データ転送に遅延が発生した時の不足を補う方法が採用される。

【0012】ここで、端末202側に受信バッファ207を設けるということは、結局、図19(B)のバッファ遷移において、バッファ占有量の上限を、デコーダバッファ208の規格である 224 KByte から受信バ

バッファ207による蓄積量の分だけかさ上げするのとおむね等価である。図21(A)、(B)に、受信バッファ297を追加する前後のバッファ占有量を並べて示す。なお、図21(A)に示されているのは、図19(B)と同一のバッファ遷移である。

【0013】受信バッファ207の追加によって、バッファ遷移の許容範囲が拡がり、その結果、図19(B)のバッファ遷移、すなわち図21(A)のバッファ遷移は、図21(B)のようになって、ネットワークの転送レートが低下しても、アンダーフローを回避することが可能となる。反面、`v b v_d e l a y`時間が、受信バッファ207による蓄積量に相当する時間だけ長くなり、デコーダ209でのデコード開始および再生装置212での再生開始が遅れる。つまり、頭出し時間が、受信バッファ207へのデータ蓄積にかかる時間の分だけ長くなる。

#### 【0014】

【発明が解決しようとする課題】以上から明らかなように、小規模LANなどの信頼性や伝送速度の保証されたネットワーク環境において、MPEG等のマルチメディアデータをストリーミング再生する場合には、基本的に、コーデックの規格で定められた再生初期遅延時間(`v b v_d e l a y`)やデコーダバッファ遷移をきちんと遵守するようなシステム設計になってさえいれば、デコーダバッファのアンダーフロー オーバーフローが起こってストリーミング再生が破綻をきたすことはない。

【0015】しかしながら、インターネットなどの広域ネットワーク環境においては、通信経路の伝送特性変動に伴なう転送ジッタが無視できないほど大きいため、従来の端末202は、コーデックの規格で定められたデコーダバッファ(`v b v`バッファ)に加えて、図20の受信バッファ207のような、転送ジッタ吸収のためのバッファを持つ場合が多い。このとき、次のような課題が存在する。

【0016】端末に搭載されるジッタ吸収用のバッファの容量は、一般に、機種によって様々である。そのため、同じデータを同じ条件下で配信しても、バッファ容量の多い機種ではストリーミング再生を破綻なく行えるが、少ない機種では、ジッタを吸収しきれずに破綻する場合があった。

【0017】この課題を解決するには、例えば、端末の搭載メモリ量を増やして、ジッタ吸収用のバッファ容量を十分確保すればよい。しかしながら、搭載メモリ量は、端末の価格を決める主な要因の一つであり、可能な限り少なく抑えたい要求がある。加えて、ジッタ吸収用のバッファ容量が多すぎると、再生開始までの頭出し時間が長くなってしまい、ユーザーにいらだちを感じさせてしまうという新たな問題が発生する。

【0018】それゆえに、本発明の目的は、端末のバッ

ファ容量が機種によって異なっていても、ネットワークの伝送能力が変動しても、バッファのアンダーフロー オーバーフローによるストリーミング再生の破綻を回避することが可能であり、しかも、ストリーミング再生の破綻回避と、頭出し時の待ち時間短縮とを互いに両立させることができるようにストリーミング方法を提供することである。

#### 【0019】

【課題を解決するための手段および発明の効果】第1の発明は、サーバが端末へネットワークを通じてストリームデータを送信し、かつ端末が当該ストリームデータを受信しつつ再生するストリーミング方法であって、端末が、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定ステップ、当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、端末が、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定ステップ、決定した目標時間および遅延時間を、端末がサーバに通知するステップ、サーバが端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御ステップを備える。

【0020】上記第1の発明では、端末が、自身のバッファ容量とネットワークの伝送能力とに応じた目標量を決定し、さらに、バッファ容量を伝送能力で除して得られる値を超えない範囲内で、遅延時間を決定する。サーバは、こうして端末が決定した目標量および遅延時間に基づいて送信速度を制御するので、端末のバッファ容量が機種によって異なっていても、ネットワークの伝送能力が変動しても、バッファ量および伝送能力に応じた送信速度制御が行え、その結果、バッファのアンダーフロー オーバーフローによるストリーミング再生の破綻を回避することが可能となる。しかも、目標量とは独立に遅延時間が決定されるので、ストリーミング再生の破綻回避と、頭出し時の待ち時間短縮とを互いに両立させることができる。

【0021】ここで、遅延時間が、バッファ容量を伝送能力で除して得られる値以下に制限されるのは、遅延時間がこの値を超えると、ストリーミング再生の破綻が起こる恐れがあるためである。この値を超えない範囲であれば、遅延時間をどのような値に決めてよい。ただし、値を決める際には、伝送能力の変動に対する耐性と、頭出し時の待ち時間との間のバランスが考慮される。

【0022】第2の発明は、第1の発明において、制御ステップにおいて、サーバは、端末のバッファに蓄積されているストリームデータの量が、当該目標量の近傍において当該目標量を超えることなく遷移するように、当

該送信速度を制御することを特徴とする。

【0023】上記第2の発明では、蓄積量が目標量の近傍において目標量を超えることなく遷移するので、バッファのアンダーフロー やオーバーフローが起こりにくい。

【0024】第3の発明は、第2の発明において、制御ステップにおいて、サーバは、当該送信速度と、当該遅延時間と、端末がストリームデータをデコードする速度とに基づいて、端末のバッファに蓄積されるストリームデータの量を予測算出することを特徴とする。

【0025】上記第3の発明では、サーバが蓄積量を予測算出して、その量に基づいて送信速度制御を行うので、蓄積量を目標量の近傍で目標量を超えないように遷移させることができる。

【0026】ここで、端末が現在の蓄積量をサーバに通知し、サーバは、通知に基づいて送信速度制御を行ってもよい。しかし、この場合、端末からサーバへの情報伝達に時間がかかるので、サーバは、過去の蓄積量に基づいて送信速度制御を行うことになる。そのため、蓄積量を目標量の近傍で目標量を超えないように遷移させることができるとは限らない。

【0027】第4の発明は、第1の発明において、端末が、ネットワークの伝送能力が所定の閾値を跨いで変化したことを検出する検出ステップ、検出ステップでの検出結果に応じて、端末が当該目標量を変更する目標量変更ステップ、および変更後の目標量を、端末がサーバに通知するステップをさらに備え、制御ステップにおいて、サーバは、変更後の目標量の通知を受けると、端末のバッファに蓄積されるストリームデータの量が、当該変更後の目標量の近傍において当該変更後の目標量を超えることなく遷移するように、当該送信速度を制御することを特徴とする。

【0028】上記第4の発明では、伝送能力が閾値を跨いで変化すると、端末によって目標量が変更される。サーバは、変更後の目標量の近傍において変更後の目標量を超えることなく遷移するように送信速度を制御して、目標量の変更に追従する。

【0029】第5の発明は、第4の発明において、検出ステップでネットワークの伝送能力が第1の閾値を跨いで低下したことを検出すると、端末は、目標量変更ステップにおいて、当該目標量を増加させる向きに変更し、制御ステップにおいて、サーバは、当該目標量が増加されたのに応じて、当該送信速度を上昇させる向きに制御することを特徴とする。

【0030】上記第5の発明では、伝送能力が第1の閾値を跨いで変化すると、端末によって目標量が増加される。サーバは、送信速度を上昇させることにより、目標量の増加に追従する。

【0031】第6の発明は、第5の発明において、当該第1の閾値は、実現可能な最大の伝送能力と、ストリー

ムデータの転送ロスが発生し始めるような伝送能力との略中間の値であることを特徴とする。

【0032】上記第6の発明では、伝送能力が低下しつつあるとき、ストリームデータの転送ロスが発生し始める前に、送信速度を上昇させて蓄積量を増やしておく。これにより、伝送能力の低下が進行したときに、ストリーミング再生が破綻するのを防ぐことができる。

【0033】第7の発明は、第4の発明において、検出ステップでネットワークの伝送能力が当該第1の閾値より小さい第2の閾値を跨いで低下したことを検出すると、端末は、目標量変更ステップにおいて、当該目標量を減少させる向きに変更し、制御ステップにおいて、サーバは、当該目標量が減少方向に変更されたのに応じて、当該送信速度を低下させる向きに制御することを特徴とする。

【0034】上記第7の発明では、伝送能力が第2の閾値を跨いで変化すると、端末によって目標量が減少される。サーバは、送信速度を低下させることにより、目標量の減少に追従する。

【0035】第8の発明は、第7の発明において、当該第2の閾値は、ストリームデータの転送ロスが発生し始めるような伝送能力と対応する値であることを特徴とする。

【0036】上記第8の発明では、伝送能力の低下が進行して、ストリームデータの転送ロスが発生し始めるとき、一転、送信速度を低下させる。失われたデータの再送処理を妨害しないためである。

【0037】ここで、送信速度を低下させる場合、サーバは、低下幅に応じた頻度でフレームの送信をスキップしなければならない。フレームがスキップされると、端末が再生して得られる映像や音声の品位劣化が起こる。この品位劣化を抑えるために、下記第9の発明では、スキップされるフレームとして、再生時刻に間に合わないフレームが選択される。下記第10の発明では、スキップされるフレームとして、重要度の低いフレームと、重要度は高いが再生時刻に間に合わないようなフレームとが選択される。

【0038】第9の発明は、第8の発明において、目標量変更ステップにおいて、端末が当該目標量を減少させる向きに変更すると、制御ステップにおいて、サーバは、送信しようとするストリームを構成する各フレームの再生時刻を現在時刻と逐次比較して、再生時刻が現在時刻よりも古いフレームの送信をスキップし、それによって当該送信速度を低下方向に制御することを特徴とする。

【0039】上記第11の発明では、再生時刻に間に合わないフレームが選択的にスキップされるので、無作為的にスキップするのと比べて、送信速度の低下による品位劣化を少なく抑えることができる。

【0040】第10の発明は、第8の発明において、目

標量変更ステップにおいて、端末が当該目標量を減少させる向きに変更すると、制御ステップにおいて、サーバは、送信しようとするストリームを構成する各フレームの重要度を基準値と逐次比較して、重要度が基準値未満であるフレームについては、全て送信をスキップし、重要度が基準値以上であるフレームについては、それぞれの再生時刻を現在時刻と逐次比較して、再生時刻が現在時刻よりも古いものだけ送信をスキップし、それによって当該送信速度を低下方向に制御することを特徴とする。

【0041】上記第10の発明では、重要度の低いフレームと、重要度は高いが再生時刻に間に合わないようなフレームとが選択的にスキップされるので、無作為的にスキップするのと比べて、送信速度の低下による品位劣化を少なく抑えができる。

【0042】ここで、第10の発明のような、スキップすべきフレームを選択する際に再生時刻に間に合うか否かに加えて重要度を考慮する方法は、典型的には、MPEGによる映像フレームに対して用いられる。この場合、送信速度を低下させるとき、PやBのフレームが重要度の低いフレームとしてスキップされる一方、Iフレームは、重要度の高いフレームとして、再生時刻に間に合わない場合を除いてスキップされることがないので、送信速度低下による再生画像の品位劣化が最小限に抑えられる。なお、MPEGによる音声フレームの場合、フレーム間に重要度の違いがないので、再生時刻に間に合うか否かだけを考慮すればよい。

【0043】第11の発明は、ネットワークを通じてストリームデータを送信するサーバと、当該ストリームデータを受信しつつ再生する端末とからなるシステムであって、端末は、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定手段、当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定手段、および決定した目標時間および遅延時間をサーバに通知する手段を備え、サーバは、端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御手段を備える。

【0044】第12の発明は、ネットワークを通じてストリームデータを送信するサーバと共に用いられ、当該ストリームデータを受信しつつ再生する端末であって、サーバには、端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御手段が備わり、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目

標量を決定する目標量決定手段、当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定手段、および決定した目標時間および遅延時間をサーバに通知する手段を備える。

【0045】第13の発明は、ストリームデータを受信しつつ再生する端末と共に用いられ、ネットワークを通じて当該ストリームデータを送信するサーバであって、端末には、自身のバッファ容量とネットワークの伝送能力とに関連して、自身のバッファに蓄積すべきストリームデータの目標量を決定する目標量決定手段、当該バッファ容量を当該伝送能力で除して得られる値を超えない範囲内で任意に、自身のバッファにストリームの先頭データを書き込んでから当該データを読み出して再生開始するまでの遅延時間を決定する遅延時間決定手段、および決定した目標時間および遅延時間をサーバに通知する手段が備わり、端末へネットワークを通じてストリームデータを送信する際に、通知された目標量および遅延時間に基づいて送信速度を制御する制御手段を備え、制御手段は、端末のバッファに蓄積されているストリームデータの量が、当該目標量の近傍において当該目標量を超えることなく遷移するように、当該送信速度を制御することを特徴とする。

【0046】第14の発明は、上記第1の発明のようなストリーミング方法を記述したプログラムである。

【0047】第15の発明は、上記第14の発明のようなプログラムを記録した記録媒体である。

【0048】

【発明の実施の形態】以下、本発明の実施形態について、図面を参照しながら説明する。図1は、本発明の一実施形態に係るストリーミング方法を実行するサーバ・クライアント・システムの構成例を示すブロック図である。図1において、本システムは、サーバ101と、そのクライアントとして動作する端末102とを備えている。サーバ101側には、映像や音声のデータが蓄積されている。このデータは、MPEGによって符号化圧縮されたデータである。サーバ101は、端末102からの要求に応じ、蓄積しているデータをパケット化してストリームを生成する。そして、ネットワーク103を通じ、生成したストリームを端末102に送信する。端末102は、サーバ101から送信されるストリームを受信してデコードし、得られた映像や音声を表示出力する。

【0049】図2は、図1のサーバ101の構成を示すブロック図である。図2において、サーバ101は、蓄積デバイス411と、送受信モジュール402と、生成モジュール405と、RAM404と、CPU412と、ROM413とを備えている。蓄積デバイス411

には、映像や音声のデータが蓄積されている。この蓄積デバイス411内のデータが生成モジュール405に与えられる。生成モジュール405は、読み出しバッファ407と、パケット生成回路406と、パケット生成バッファ408とを含み、与えられるデータをパケット化してストリームを生成する。

【0050】送受信モジュール402は、ネットワークコントローラ410と、送信バッファ409とを含み、生成モジュール405によって生成されたストリームを端末102へ、ネットワーク103経由で送信する。また、端末102からネットワーク103経由で送信されてくる情報を受信する。

【0051】送受信モジュール402によって受信された端末102からの情報は、RAM404に書き込まれる。ROM413には、サーバ制御プログラムが格納されており、CPU412は、RAM404に記憶されている端末からの情報を参照しつつROM413内のプログラムを実行し、それによって、送受信モジュール402および生成モジュール405の制御を行う。なお、ここではプログラムがROM413に格納されているとしたが、ROM以外の記憶媒体、例えばハードディスクやCD-ROM等に格納されていてもよい。

【0052】図3は、図1の端末102の構成を示すブロック図である。図3において、端末102は、送受信モジュール507と、再生モジュール510と、表示デバイス511と、ROM502と、CPU503とを備えている。送受信モジュール507は、ネットワークコントローラ506と、受信バッファ505とを含み、サーバ101からネットワーク103経由で送信されてくるストリームを受信する。また、CPU503からの情報をサーバ101へ、ネットワーク103経由で送信する。

【0053】送受信モジュール507によって受信されたストリームが、再生モジュール510に入力される。再生モジュール510は、デコーダバッファ508と、デコーダ509とを含み、入力されるストリームをデコードして再生する。再生モジュール510により再生されたデータが表示デバイス511に与えられ、表示デバイス511は、そのデータを映像に変換して表示する。

【0054】ROM502には、端末制御プログラムが格納されており、CPU503は、ROM502内のプログラムを実行し、それによって、送受信モジュール507、再生モジュール510および表示デバイス511の制御を行う。

【0055】以上のように構成されたシステムの動作を、以下に説明する。図4は、図1のシステムの全体動作を説明するためのシーケンス図である。図4には、サーバ101側の送受信層および制御層と、端末102側の送受信層および制御層とが示されており、これら各層の間でやりとりされるコマンドやストリームが時系列的に並べられている。

【0056】最初、本システムの全体的な動作について、図4を用いて説明する。図4において、最初、端末102からサーバ101へ、コマンド"SETUP"が送信される。サーバ101では、"SETUP"に応じて初期設定が行われ、設定が完了すると、サーバ101から端末102へ、"OK"が応答される。

【0057】サーバ101から"OK"が返ってくると、端末102からサーバ101へ、コマンド"PLAY"が送信される。サーバ101では、送信準備が行われ、準備が完了すると、サーバ101から端末102へ、"OK"が応答される。

【0058】サーバ101から"OK"が返ってくると、端末102は、ストリームを待ち受ける態勢へと移行する。この"OK"の応答に引き続いて、サーバ101は、ストリームの送信を開始する。

【0059】その後、端末102からサーバ101へ、コマンド"TEARDOWN"が送信され、サーバ101は、"TEARDOWN"に応じてストリーム送信を終了する。送信が終了されると、サーバ101から端末102へ、"OK"が応答される。サーバ101から"OK"が返ってくると、端末102は、ストリーム待ち受け態勢から脱する。

【0060】以上が本システムの全体的な動作の概要であり、上で説明した限りでは、本システムの動作は、従来と同様である。本システムの動作が従来と異なるのは、次の(1)および(2)の2点である。

(1) 端末102からサーバ101へのコマンド"SETUP"にパラメータ"S\_target"および"T\_delay"が添付されており、サーバ101は、ストリームを送信する際、これらのパラメータに基づいて送信速度を制御する。

【0061】上記(1)において、"S\_target"は、端末102がバッファに蓄積するデータ量の目標値であり、その端末102に備わるバッファ(図3の例では、受信バッファ505およびデコーダバッファ508)の総容量("S\_max")と、ネットワーク103の伝送能力とに基づいて決定される。従って、"S\_target"は、一般に、端末102の機種によって値が異なる。

【0062】また、"T\_delay"は、端末102が先頭データをバッファに書き込んでから、そのデータを読み出してデコードを開始するまでの時間(つまり頭出し遅延時間)であり、"S\_target"を送信速度(後述)で除して得られる値を最大値として、その最大値を超えない範囲内で任意の値に決定される。すなわち、「"S\_target"を送信速度で除して得られる値を超えない範囲内で」という条件が付くものの、端末102は、"S\_target"とは独立に"T\_delay"を決めることができる。

【0063】また、「送信速度」は、単位時間に送信する情報の量をいい、例えば、単位時間に送信するパケットの個数が決められている場合、各パケットに詰め込むデータの量を増減させることによって、送信速度を制御することができる。また、各パケットに詰め込むデータの量が決められている場合、パケットと次のパケットとの時間的な間隔を伸縮させることによって、送信速度を制御することができる。あるいは、両方を同時に使う（すなわち、各パケットに詰め込むデータの量を増減させ、かつパケットと次のパケットとの時間的な間隔を伸縮させる）ことによっても、送信速度を制御することができる。本実施形態では、各パケットに詰め込むデータの量を増減させることにより速度制御を行うものとする。

【0064】(2) 端末102は、ストリームの配信中、必要に応じて“S\_target”を変更することができる。この場合、変更後の“S\_target”が端末102からサーバ101に通知され、以降、サーバ101は、変更後の“S\_target”に基づいて送信速度を制御する。

【0065】上記(2)において、“S\_target”的変更は、ネットワーク103の伝送能力の変動に応じて行われる。具体的には、端末102が携帯電話の場合、電界強度（例えば「強・中・弱・圏外」の4段階の強度）を検知することができるので、この電界強度の変化を「ネットワーク103の伝送能力の変動」と見なしして、“S\_target”を変化させる。例えば、電界強度が「強」から「中」に変化すると、端末102は、“S\_target”的値をより大きな値に変更し、「中」から「弱」に変化すると、“S\_target”的値をより小さな値に変更する。本システムの動作が従来と異なるのは、主として上の2点である。

【0066】次に、本システムの全体動作の具体例を詳細に説明する。図4において、端末102は、ストリーム再生を開始するのに先立ち、端末制御プログラムに従ってCPU503がROM502より端末に固有のパラメータ群を抽出する。このパラメータ群中には、受信バッファ505とデコーダバッファ508とを合わせた総容量（端末102が実際に蓄積できる最大データ量）S\_maxが含まれる。一方、CPU503は、ストリーム再生補助データなどの事前入手の手続きによって、受信したいストリームデータの符号化圧縮レートVrや、ビデオないしオーディオのフレーム発生周期Tfrmを知っているものとする。また、CPU503は、ネットワークインターフェースを通じ、ネットワーク103の伝送能力、たとえば携帯電話における受信電波強度や、通信速度（PHSの場合では64Kbps接続ないし32Kbps接続などの情報）も検出しているものとする。

【0067】CPU503は、これらS\_max、V

r、Tfrm、ネットワーク103の伝送能力（例えば有効転送速度=networkRate）などをもとに、端末102内のバッファにどれだけのデータを蓄積するかを示す目標量S\_target、およびストリーム再生を始めるまでのプレバッファリング時間（すなわち頭出し遅延時間）T\_delayを決定する。

【0068】ここで、S\_targetの本質的な意味は、今から開始するストリーミング再生において、S\_target近傍かつそれを越えないように端末の蓄積バッファ量が遷移すれば、途切れなく正常にストリーミング再生が持続できるような基準値のことである。前述のように、T\_delayが大きいと頭出し遅延時間が長くなるが、ネットワーク103の転送ジッタに対しては強くなる。しかし、遅延時間があまり長いとサービス仕様として不適切なので、T\_delayを決める際には、転送ジッタに対する耐性と、頭出し時の待ち時間との間のバランスをとる配慮が必要である。

【0069】なお、T\_delayの代わり、もしくはT\_delayと併せて、端末102内のバッファにデータを何バイトまで充填したらデコードを開始するかという充填量S\_delayを決定してもよい。端末102がS\_delayのみを決定してサーバ101に通知する場合は、サーバ101側でT\_delay=S\_delay/networkRateなる式を用いて、S\_delayをT\_delayに換算することが可能である。また、S\_delayの値は、バッファ総量S\_maxに対する充填率rS(%)であってもよい（この場合、換算式は、S\_delay=S\_max \* rS / 100となる）。

【0070】端末102は、S\_targetと、T\_delay（および／またはS\_delay）とを準備すると、図4に示されているように、サーバ101に対し、ストリーム配信の準備を促すSETUPコマンドを発行する。SETUPコマンド中には、引数としてS\_targetと、T\_delay（および／またはS\_delay）とが含まれている。サーバ101は、SETUPを受信すると、引数をRAM404に記憶して、ストリーム配信のための初期設定を行う。具体的には、サーバ101のCPU412が、最初メモリ404から引数を取り出し、次いで、たとえばストリームのソースファイルを蓄積デバイス411から読み出してバッファ407に書き込む操作と、読み出したデータをパケット化するパケット生成回路406のパラメータ設定とを行う。なお、パケット生成回路406は、必ずしも専用のハードウェアである必要はなく、サーバ101（たとえばワークステーションなどで実現される）のCPU412に同様のパケット化処理を実行させるプログラム（ソフトウェアアルゴリズム）であってもよい。

【0071】前述のS\_targetと、T\_delay（および／またはS\_delay）との2つの値も、

パケット生成回路406に引き渡される。パケット生成回路406では、これらの値を用いて最適なレート制御パラメータの算出が行われ、その結果、端末102へのストリーム配信に適したレートでパケットが生成、送出されるようになる。ネットワーク103中にパケットを送出する準備が正常に完了すると、図4のように、送受信層から制御層にOKが返り、次いで、端末102へ向けてSETUPコマンドに対するOKが返る。こうして、本システムにおいて、ストリーム配信準備が完了する。

【0072】次いで、端末102がサーバ101に対し、ストリーム配信の開始を促すPLAYコマンドを発行する。サーバ101は、PLAYを受信すると、ストリームデータの配信を開始する。端末102は、サーバ101からのストリームデータを受信して蓄積する。そして、蓄積開始から前述のプレバッファリング時間( $T_{delay}$ )が経過したのちに、ストリームデータのデコード、再生を開始する。このときストリーム配信は、SETUP時に設定された適切なレート制御パラメータに基づいてなされていることはいうまでもない。

【0073】ストリーム再生の終了時には、端末102よりサーバ101に対し、TEARDOWNコマンドが発行される。サーバ101は、TEARDOWNを受信すると、ストリーム配信の終了処理を行い、全手続きを完了させる。以上が、本システムの具体的な動作例である。

【0074】以下には、端末102の動作について詳細に説明する。端末102は、インターネットに接続可能な携帯電話であり、電界強度(受信電波強度)を検知する機能を持っているとする。図5は、図1の端末102の動作を示すフローチャートである。図5において、最初、端末102は、2つのパラメータ $S_{target}$ および $T_{delay}$ の値を決定する(ステップS101)。

【0075】ここで、上記ステップS101の処理内容を具体的に説明する。図6は、図3のROM502の記憶内容を示す図である。図6に示すように、ROM502内には、端末制御プログラムと、電界強度および $S_{target}$ が互いに対応付けて記載されたテーブル601と、パラメータ $T_{delay}$ の値とが記憶される。パラメータ $S_{target}$ の値としては、電界強度「強」と対応する $S_{target}1$ 、「中」と対応する $S_{target}2$ 、「弱／圏外」と対応する $S_{target}3$ の3つの値が記憶されている。一方、パラメータ $T_{delay}$ の値は、1つだけが記憶されている。

【0076】上記3つの値 $S_{target}1\sim3$ は、次の関係を満たすように決められる。

$$S_{target}3 < S_{target}1 < S_{target}2 \leq S_{max}$$

一方、値 $T_{delay}$ は、値 $S_{max}$ をネットワーク103の実効的な伝送能力で除して得られる値を超えないように決められる。

【0077】一例として、 $S_{max}$ が512(KB)であれば、 $S_{target}1=256$ (KB),  $S_{target}2=384$ (KB),  $S_{target}3=128$ (KB)などのように決められる。また、ネットワーク103の実効的な伝送能力を384(Kbps)すなわち48(KB/sec)とすると、 $T_{delay}$ は、 $512 \div 48 = 10.7$ (秒)を超えない範囲で、任意の値(例えば4秒や3秒など)に決定される。

【0078】上記ステップS101では、ROM502から、初期値としての $S_{target}1$ と、値 $T_{delay}$ とが読み出される。

【0079】なお、ここでは、 $S_{target}1\sim3$ と、 $T_{delay}$ の値とが予め計算されてROM502に記憶されており、CPU503は、必要な値をROM502から読み出すようにしているが、代わりに、バッファの総容量と、ネットワーク103の実効的な伝送能力と、 $S_{target}$ および $T_{delay}$ の値を計算するためのプログラムとをROM502に記憶しておいてよい。この場合、CPU503は、必要があれば、その都度、ROM502から容量、速度およびプログラムを読み出して、 $S_{target}$ および $T_{delay}$ の値を計算する。また、ここでは、 $T_{delay}$ の値が1つだけであるが、複数の値を準備しておいて、その中から選択してもよい。以上がステップS101の処理内容である。

【0080】再び図5において、端末102は、ステップS101で決定された $S_{target}$ および $T_{delay}$ をSETUPコマンドに添付して、サーバ101へ向けて送信する(ステップS102)。応じて、サーバ101からストリームが送られてくる。ストリーム送信の際、サーバ101は、端末から通知された $S_{target}$ および $T_{delay}$ に基づく送信速度制御を実行する(サーバ側の動作については後述)。

【0081】次に、端末102は、サーバ101から送られてくるストリームを受信して、バッファに書き込む動作を開始する(ステップS103)。具体的には、図3に示されているように、ネットワーク103を通じて送られてくるストリームは、まずネットワークコントローラ506を経由して受信バッファ505に書き込まれる。時間が経過して受信バッファ505が満杯になると、受信バッファ505内のストリームが先頭データから順番に読み出されて、デコーダバッファ508へと書き込まれていく。

【0082】次に、端末102は、バッファリング開始から時間が $T_{delay}$ だけ経過したか否かを判定し(ステップS104)、その判定結果が否定であれば、

肯定となるまで待機する。ステップS104の判定結果が肯定となると、端末102は、バッファからストリームを読み出してデコード・再生する動作を開始する（ステップS105）。具体的には、図3において、CPU503がバッファリング開始からの経過時間を計測しており、その計測結果がROM502内のT\_delayと一致した瞬間、再生モジュール510に命じて、デコーダバッファ508内のストリームを先頭データから順番に読み出してデコーダ509に入力する処理を開始させる。

【0083】次に、端末102は、ネットワーク103の伝送能力が閾値を跨いで変化したか否かを判定する（ステップS106）。この判定は、具体的には、次のようにして行われる。例えば、ネットワーク103を管理するホストコンピュータ（図示せず）が、ネットワーク103の伝送能力に関する情報をネットワーク103経由で端末102に随時配信するようにし、端末102は、ホストコンピュータからの情報をもとに変化の有無を判定する。

【0084】この場合、具体的には、図3に示すように、伝送能力に関する情報が送受信モジュール507を通じてCPU503へと送られる。ROM502には、予め閾値が格納されており、CPU503は、送られてきた情報と、保持している前回の情報と、ROM502内の閾値とを互いに比較することにより、伝送能力が閾値を跨いで変化したか否かを判定することができる。

【0085】または、ネットワーク103を管理するホストコンピュータがその伝送能力に関する情報を端末102に配信する機能を持たない場合、端末102は、例えば、次のようにして自ら判定を行うことができる。すなわち、端末102が携帯電話の場合、図7（後述）に示すように、周囲の電界強度を検知して、検知結果を「強・中・弱・圏外」のように表示する機能を持っている。この電界強度の変化をネットワーク103の伝送能力の変化と見なせば、端末102は、検出を簡単に行えることになる。

【0086】ステップS106の判定結果が肯定の場合、端末102は、新たなS\_targetを決定し（ステップS107）、それをサーバ101へ向けて送信する（ステップS108）。一方、ステップS106の判定結果が否定の場合、ステップS107、S108をスキップして、ステップS109（後述）を実行する。

【0087】ここで、上記ステップS106およびステップS107の処理内容について、詳しく説明する。以下では、端末102が携帯電話であり、電界強度の変化に応じてS\_targetを変更する場合を説明する。図7は、あるエリアにおける電界強度の分布と、端末の移動に伴う伝送能力の変化とを示す模式図である。図7（A）には、3つの中継局B1～B3を含むエリアにお

ける電界強度分布が示されている。図7（A）において、各中継局B1～B3を中心とする同心円が、互いに等しい電界強度の点を繋いでできる等電界曲線である。

【0088】例えば、中継局に最も近い同心円703内では、電界強度が「強」であり、この同心円703から次の同心円704までの間の領域では「中」となる。さらに、同心円704から同心円705までの間では「弱」、同心円705の外側の領域では「圏外」となる。ただし、各中継局を中心とする同心円は、一部が互いに交差しており、電界強度が「圏外」となる領域は、わずかしかない。

【0089】いま、端末102は、矢印702で示される経路に沿って、中継局B1の近傍から中継B2の近傍へ移動しようとしている。図7（B）には、図7（A）の矢印702に沿った電界強度（これをネットワーク103の伝送能力と見なすことができる）が示されている。図7（B）に示されているように、電界強度は、端末102が中継局の近傍にあるとき「強」であり、中継局B1から離れるにつれて「中」、「弱」、「圏外」のようにだんだん弱くなっていく。そして、中継局B1の「圏外」となった直後、端末102は、中継局B2の「圏内」に入り、電波強度が「弱」、「中」、「強」のようにだんだん強くなっていく。

【0090】上記のように移動する端末102は、電界強度が「強」から「中」へと変化した瞬間、ネットワーク103の伝送能力が閾値Aを跨いで変化したと判定して、新たなS\_targetを決定し、「中」から「弱」へと変化した瞬間、伝送能力が閾値Bを跨いで変化したと判定して、新たなS\_targetを決定する。逆に、「弱」から「中」へと変化した瞬間、伝送能力が閾値Bを跨いで変化したと判定して、新たなS\_targetを決定し、「中」から「強」へと変化した瞬間、伝送能力が閾値Aを跨いで変化したと判定して、新たなS\_targetを決定する。

【0091】なお、一般的には、閾値Aは、ネットワーク103において実現可能な最大の伝送能力と、ストリームの転送ロスが発生し始めるような伝送能力との略中間の値である。閾値Bは、ストリームの転送ロスが発生し始めるような伝送能力と対応する値である。

【0092】新たなS\_targetは、ROM502内のテーブル601（図6）を参照することにより、次のように決定される。図8は、図5のステップS107の詳細を示すフローチャートである。図8において、端末102は、最初、変化後の電界強度が「強」か否かを判定し（ステップS201）、判定結果が肯定であれば、新たなS\_targetをS\_target1に決定する（ステップS202）。ステップS201の判定結果が否定あれば、ステップS202をジャンプして、ステップS203に進む。

【0093】次に、端末102は、変化後の電界強度が

「中」か否かを判定し(ステップS203)、判定結果が肯定であれば、新たなS\_targetをS\_target2に決定する(ステップS204)。ステップS203の判定結果が否定であれば、ステップS204をジャンプして、ステップS205に進む。

【0094】次に、端末102は、変化後の電界強度が「弱／圏外」か否かを判定し(ステップS205)、判定結果が肯定であれば、新たなS\_targetをS\_target3に決定し(ステップS206)、その後、図5のフローに戻る。ステップS205の判定結果が否定であれば、ステップS206をジャンプして、図5のフローに戻る。

【0095】従って、図7(A)の矢印702に沿って端末102が移動して行く場合、電界強度の変化に伴って、端末102は、パラメータS\_targetの値を、S\_target1→S\_target2→S\_target3→S\_target2→S\_target1のように変化させる。具体例を挙げれば、256(KB)→384(KB)→128(KB)→384(KB)→128(KB)のように変化させる。以上が、ステップS106およびステップS107の具体的な処理例である。

【0096】再び図5において、ステップS108で端末102が新たなS\_targetをサーバ101へ向けて送信すると、それに応じて、サーバ101は、パラメータS\_targetの値を、端末102から新たに通知された値に変更して、送信速度制御を続行する。

【0097】次に、端末102は、ストリーミング再生を終了するか否かを判断し(ステップS109)、終了する場合は、サーバ101へコマンドTEARDOWNを送信すると共に、ストリームの受信およびバッファリングを停止し(ステップS110)、次いで、再生処理を停止する(ステップS111)。一方、ストリーミング再生を継続する場合には、端末102は、ステップS106に戻って、上記と同様の処理を繰り返す。以上が、端末102の動作である。

【0098】次に、サーバ101の動作について詳細に説明する。なお、ここでは説明を簡単にするために、サーバ101は、MPEG1ビデオ(ISO/IEC 1172-2)、MPEG2ビデオ(ISO/IEC 13818-2)、あるいはMPEG2-AACオーディオ(ISO/IEC 13818-7)のような、固定周期Tfrmでフレームを発生させる符号化圧縮アルゴリズムを用いて符号化を行い、かつ、固定周期Tsで符号化データのパケット化を行うものとする。このパケット化は、フレーム単位で行われるものとする。

【0099】最初、サーバ101が行うストリーム送信速度制御の概要について、図9～図11を用いて説明する。図9～図11は、サーバ101が行うストリーム送信速度制御によって、端末102のバッファに蓄積され

ているデータ量(バッファ占有量)がどのように遷移するかを示す図である。サーバ101は、送信先の端末102において、バッファ占有量が図9～図11に示されているごとく遷移するように、ストリームの送信速度を制御する。

【0100】図9には、バッファ占有量がS\_targetに近づいていく様子が示されている。図10には、バッファ占有量がS\_targetの近傍で遷移している状態で、S\_targetの値がより大きな値(S\_target2)に変更された場合に、バッファ占有量がS\_target2に近づいていく様子が示されている。図11には、バッファ占有量がS\_targetの近傍で遷移している状態で、S\_targetの値がより小さな値(S\_target3)に変更された場合に、バッファ占有量がS\_target3に近づいていく様子が示されている。

【0101】図9～図11に共通して、"S\_max"は、端末102のバッファの総容量であり、"Sum"が、バッファ占有量である。"delta(0, 1, 2, …)"は、サーバ101が単位時間Tsあたりに送信するデータの量(すなわち、1つのパケットに詰め込まれているデータの量)を示す。ここで、単位時間Tsは、サーバ101がパケットを送信する周期であり、固定値である。"L(0, 1, 2, …)"は、1つのフレームあたりのデータ量である。

【0102】サーバ101は、端末102からT\_delayの値の通知を受けると、その値に基づいてストリームの送信速度を制御する。この速度制御は、1つのパケットに詰め込むデータの量を変化させることにより行われる。

【0103】図9において、サーバ101が最初に送信したパケット(i=0)には、量delta0のデータが詰め込まれており、時刻t=0では、バッファ占有量Sumは、delta0となる。単位時間Tsが経過すると、次のパケット(i=1)が送られてくるが、そこには、量delta1のデータが詰め込まれている。従って、時刻t=Tsでは、Sumは、{delta0+delta1}となる。以降、単位時間Tsが経過する毎に、次々とパケット(i=2, 3, …)が送られてきて、Sumにdelta2, delta3, …が加算されていく。

【0104】一方、3つ目のパケット(i=2)が送られてくる以前である時刻t=T\_delayに、バッファからデータを読み出してデコードする処理が開始される。デコードはフレーム単位で行われるので、t=T\_delay以降、固定周期Tfrm毎に、SumからL0, L1, L2…が減算されていく。

【0105】すなわち、バッファ占有量Sumは、時刻t=0以降、周期Ts毎に、delta0, delta1, …が加算されて、だんだん増加していく。その一方

で、時刻  $t = T_{delay}$  以降、周期  $T_{frm}$  每に  $L_0, L_1, L_2 \dots$  が減算されていく。従って、 $Sum$  が目標値  $S_{target}$  に達する直前までの期間は、1つのパケットに詰め込むデータ量を標準よりも多くして（より一般的には送信速度を速くして）、バッファへの書き込み速度がバッファからの読み出し速度を上回るようになり、それ以降は、1つのパケットに詰め込むデータ量を標準に戻して、書き込み速度と読み出し速度とを均衡させれば、 $Sum$  を目標値  $S_{target}$  の近傍で遷移させることが可能となる。

【0106】このような送信速度制御を行えば、図10、図11のように、目標値  $S_{target}$  が途中で新たな目標値 ( $S_{target2}, 3$ ) に変更された場合にも、 $Sum$  を新たな目標値 ( $S_{target2}, 3$ ) の近傍で遷移させることが可能となる。

【0107】すなわち、図10において、 $Sum$  が目標値  $S_{target}$  の近傍で遷移している状態で、 $S_{target}$  がより大きな値 ( $S_{target2}$ ) に変更されると、サーバ101は、以降のパケット ( $i = 3, 4$ ) に詰めるデータの量を増やすことによって、バッファへの書き込み速度がバッファからの読み出し速度を上回るようにする。 $Sum$  が新たな目標値  $S_{target2}$  に達して以降は、1つのパケットに詰め込むデータ量を標準に戻して、書き込み速度と読み出し速度とを均衡させればよい。

【0108】また、図11において、 $Sum$  が目標値  $S_{target}$  の近傍で遷移している状態で、 $S_{target}$  がより小さな値 ( $S_{target3}$ ) に変更されると、サーバ101は、以降のパケット ( $i = 3, 4$ ) に詰めるデータの量を減らすことによって、バッファへの書き込み速度がバッファからの読み出し速度を下回るようにする。 $Sum$  が新たな目標値  $S_{target3}$  に達して以降は、1つのパケットに詰め込むデータ量を標準に戻して、書き込み速度と読み出し速度とを均衡させればよい。

【0109】次に、上で説明したようなサーバ101による送信速度制御について詳細に説明する。図12は、サーバ101が行う送信速度制御アルゴリズムの一例を示すフローチャートである。図12において、最初、端末102が自身のバッファ占有量 ( $Sum$ ) を検出し、サーバ101は、端末102からバッファ占有量  $Sum$  の通知を受ける（ステップS301）。次に、サーバ101は、ステップS301で通知されたバッファ占有量  $Sum$  が、端末102から指定された目標値  $S_{target}$  の近傍で遷移しているか否かを判定する（ステップS302）。その判定結果が肯定であれば、現在の送信速度が維持される。

【0110】ステップS302の判定結果が否定の場合、サーバ101は、ステップS301で通知されたバッファ占有量  $Sum$  が目標値  $S_{target}$  よりも大

きいか否かを判定する（ステップS303）。そして、判定結果が否定であれば、送信速度を現在よりも速い速度に変更し（ステップS304）、その後、ステップS306に進む。一方、ステップS303の判定結果が肯定であれば、送信速度を現在よりも遅い速度に変更し（ステップS305）、その後、ステップS306に進む。

【0111】ステップS306では、速度制御動作を継続するか否かが判断され、判断結果が肯定の場合、ステップS301に戻って、上記と同様の動作が繰り返される。一方、判断結果が否定の場合、動作が終了される。以上が、サーバ101が行う送信速度制御の一例である。

【0112】ところで、図12の例では、端末102が自身のバッファ占有量を検出して、サーバ101に通知している。しかしその場合、端末102が検出するのは、現在時刻におけるバッファ占有量である。その上、端末102からサーバ101への情報伝達に時間がかかるので、サーバ101は、伝達遅延時間の分だけ過去のバッファ占有量に基づいて送信速度制御を行うことになり、バッファ占有量を  $S_{target}$  の近傍で遷移させるのは、現実には困難である。

【0113】これに対して、以下に説明する別の例（図13、14参照）では、未来のある時点でのバッファ占有量に基づいて送信速度制御を行うことによって、バッファ占有量を  $S_{target}$  の近傍で遷移させることができるとなる。この場合、サーバ101は、端末102から  $Sum$  の通知を受けるのでなく、未来のある時刻における端末102側のバッファ占有量  $Sum$  を予測算出する。この予測算出は、次のようにして行われる。

【0114】すなわち、図2において、ROM413には、パケット送信周期  $T_s$  (固定値) と、デコード周期  $T_{frm}$  (固定値) とが予め記憶されている。CPU412は、パケットが生成される際、そのパケットに詰め込まれたデータの量 ( $\Delta t_0, \Delta t_1, \Delta t_2, \dots$ ) をRAM404に記憶させておく。さらに、ストリームが送信される際、各フレームのデータ量 ( $L_0, L_1, L_2, \dots$ ) をRAM404に記憶させておく。

【0115】RAM404にはまた、先に端末102から通知された  $T_{delay}$  が記憶されており、CPU412は、ROM413内の  $T_s$  および  $T_{frm}$  と、RAM404内の  $\Delta t_0, \Delta t_1, \Delta t_2, \dots$  および  $T_{delay}$  を参照して所定の演算を行うことにより、未来のある時刻におけるバッファ占有量  $Sum$  を算出することができる。このような算出処理を行うことによって、サーバ101は、端末102側においてバッファ占有量  $Sum$  がどのように遷移していくか（図9～図11参照）を予測することができる。

【0116】以下、サーバ101が端末102のバッファ占有量Sumを予測算出して送信速度制御を行う具体的な動作例について、図9のバッファ遷移図、図13および図14のフローチャート、図15のパケット構成図を用いて説明する。

【0117】図9において、S\_maxは、端末102内バッファの有効蓄積量の最大値（これを簡単に「バッファの総容量」と呼んでいる）であり、S\_targetは、今回のストリーミングにおいて端末102内バッファに蓄積しようとする目標量であり、T\_delayは、頭出し遅延時間の設定値である。これら各パラメータの意味については、既に説明したとおりである。以下では、端末102よりS\_targetとT\_delayが通知されたものとして説明を行う。

【0118】本実施の形態では、理解を簡単にするために、固定時間周期Ts毎にパケットの生成・配信を行う例（i=nに相当する時刻でパケット配信：nは整数）を示している。また、i=nに相当する時刻（t=i\*Ts）でパケットの配信がなされた際に、端末102の受信バッファ505およびデコーダバッファ508内の蓄積量Sumは、数フレームに相当するデータ量が瞬時に増加しているが、これは、図15（A）に示されているように、1パケットに複数フレームを挿入するパターンでパケットを生成して端末102に配信しているためである。実際には、パケット配信には転送時間がかかり、図のように瞬時にバッファ占有量が増加する訳ではない（傾き=networkRateの斜線になる）が、あくまでモデルとして単純化して取り扱うものとする。また、時刻t=T\_delay以降、階段状にバッファ占有量が減じていくのは、その時刻に端末102でストリーム再生が始まったことを示している。すなわち、フレームの再生周期Tfrm毎に、各々のフレーム長L=L[k]（kは整数）ずつデコーダ509でデータが処理される。

【0119】図13および図14は、図9のバッファ遷移を実現するためにサーバ101によって行われる送信制御アルゴリズムの別の例を示したフローチャートである。図13がアルゴリズムの全体像であり、図14は、図13のステップS404中の関数mkPacketの一例を示すフローチャートである。このようなアルゴリズムを記述したプログラムがROM413（図2参照）に格納されており、CPU412は、このプログラムに従って各種の演算や制御を行い、その結果、図9のバッファ遷移が実現される。なお、説明を簡単にするために、ストリーム途中での配信停止などは、今回考えないものとする。以下、各ステップについて順次説明を行う。

【0120】図9において、サーバ101は、端末102から送られてきたS\_targetおよびT\_delayを受信して記憶する（ステップS401）。具体的

には、図2において、端末102からネットワーク103経由で送られてきたS\_targetおよびT\_delayの値が、ネットワークコントローラ410を通じてRAM404に書き込まれる。

【0121】なお、ここでは、端末102がパラメータS\_targetおよびT\_delayの値を決定して、結果をサーバ101に通知しているが、代わりに、サーバ101がそれらの値を予め記憶しておいてよく、あるいは、端末102の機種情報（バッファの総容量等）を記憶しておいて、この機種情報をもとにサーバ101がパラメータの値を計算してもよい。

【0122】次に、各変数の初期化が行われる（ステップS402、S403）。各変数の意味は、図14の説明にて後述する。初期化が完了すると、ステップS404以降の処理、すなわち関数mkPacketにてパケットを生成してネットワーク103中に送出する処理が開始される。生成されたパケットは、この例では固定周期Tsで端末102に配信されるので、サーバ101は、ステップS405にてタイミング調整を行ったのち、ステップS406にて送出を行う。この一連の処理が完了すると、CPU412は、関数mkPacketの実行カウンタを更新し、ステップS404に戻ってループする。ストリームデータの読み出しおよびパケット化が全て完了すると、CPU412は、関数mkPacketを抜けて、ステップS404に判定結果FALSEでreturnする。CPU412は、このとき配信が完了したと見なし、アルゴリズムを完了する。以上が、本送信制御アルゴリズムの概要である。

【0123】次に、ステップS404に示された関数mkPacketの詳細なアルゴリズムであるが、まず各変数について説明を行う。Sumは、端末102内の受信バッファ505およびデコーダバッファ508に蓄積されているデータ量の総和であり、Lは、フレームのデータ量であり、deltaは、関数mkPacketが今回呼ばれてからパケット化したデータ量の総和（すなわち1つのパケットに詰め込んだデータの量）であり、inは、蓄積デバイス411から読み出したストリームソースのフレーム数を示すカウンタであり、outは、端末102内のデコーダ509でデコードされたフレーム数を示すカウンタであり、dtsは、デコーダ509にてフレームがデコードされる時刻であり、gridは、前回の関数mkPacketの1ループを処理する際に進んだdtsの上限値である。

【0124】図14において、関数mkPacketは、大きくパケット生成アルゴリズムA1と、デコード量算出アルゴリズムA2との2つのアルゴリズムに分けられる。前者において、最初のステップ（S501）では、CPU412は、deltaをクリアする。続くステップS502では、CPU412は、L=L[in]のフレーム（既に読み出し済み）を今回のパケット生成

に用いて良いかどうかの判定を行う。判定の基準は、

(a)  $\text{Sum} + \text{L}$  を加えても  $\text{S\_target}$  を越えないこと、および (b) 今回の関数呼び出しでパケット化したデータの量 (今回 1 つのパケットに詰め込んだデータの量)  $\text{delta}$  に  $\text{L}$  を加えても、1 つのパケットに詰め込み可能なデータ量の上限値  $\text{deltaMax}$  を超えないこと、の 2 つである。

【0125】ここで  $\text{deltaMax}$  は、図 15 (A) に示される不等式

$$(\text{deltaMax} + \text{hdr}) / \text{Ts} < \text{Network Rate}$$

を満たす値であって、周期  $\text{Ts}$  以内に端末に配信可能なデータ量の最大値であり、ネットワーク 103 の実効転送レート (伝送能力) から算出が可能である。ステップ S 502 にて真と判定されると、CPU 412 は、ステップ S 503 に進み、 $L = L[\text{in}]$  のフレームをパケット化する。続くステップ S 504 では、CPU 412 は、パケット化の実行に伴い、 $\text{Sum}$  および  $\text{delta}$  を更新する。続くステップ S 505 では、CPU 412 は、次のフレームのデータを読み出しバッファ 407 から、フレーム長  $L$  を RAM 404 から、それぞれ読み出す。そして、 $L$  が 0 よりも大きいか否かを判定する。

【0126】ステップ S 505 の判定結果が否定、すなわち  $L = 0$  であれば、CPU 412 は、全データの読み出しが完了した (End of File 検出) とみなし、関数を抜ける。そして、メインフロー (図 13) のステップ S 404 に判定結果 FALSE で RETURN する。一方、判定結果が肯定、すなわち  $L > 0$  であれば、CPU 412 は、次のステップ (S 506) に進み、 $L[\text{in}]$  を配列  $\text{length}$  内に加える (RAM 404 に記憶させる)。これは後ほど説明するが、デコード量算出アルゴリズム A2 で用いるためである。次に、CPU 412 は、ステップ S 507 に進み、読み出したフレーム数カウンタ  $\text{in}$  を更新して、ステップ S 502 にループする。

【0127】上記のループによるパケット生成を繰り返すうち、 $\text{Sum}$  および  $\text{delta}$  の値がだんだん大きくなっていく。そして、ステップ S 502 にて  $\text{Sum}$  または  $\text{delta}$  が十分大きくなつたと判定されると、CPU 412 は、このループを抜けて、デコード量算出アルゴリズム A2 に入る。

【0128】デコード量算出アルゴリズム A2 において、最初のステップ (S 508) では、 $i * \text{Ts}$  が  $\text{grid}$  以上であるか否かが判定される。このステップ S 508 は、端末 102 においてデコードが開始される時刻になったかどうかを判定することが目的である。具体的には、 $\text{grid}$  が最初  $\text{T\_delay}$  に設定されているため、関数呼び出しカウンタ数  $i$  が小さくて  $t = i * \text{Ts}$  が  $\text{grid}$  未満の間は、端末 102 でのデコードが始まつてないものと判定される。図 9 では、 $i = 0$

および  $i = 1$  と対応する時刻がこれに相当する。

【0129】ステップ S 508 の判定結果が否定の場合、CPU 412 は、デコードによるフレームデータの減算を行わずに関数を抜ける。一方、 $i$  が十分大きくなつてパケット生成時刻  $t = i * \text{Ts}$  が  $\text{grid}$  以上になると、CPU 412 は、端末 102 でのデコードが既に始まつているとみなし、フレームデータの減算処理を行う。図 9 では、 $i$  が 2 以上の時刻がこれに相当する。続くステップ S 509 からステップ S 512 までのループにおいて、現在の  $\text{grid}$  時刻から次の  $\text{grid}$  時刻 ( $= \text{grid} + \text{Ts}$ ) に挟まれた時間内にデコード処理されるフレームデータの量  $\text{length}[\text{out}]$  を  $\text{Sum}$  から減算し、かつデコードしたフレーム数  $\text{out}$  をカウントアップする。

【0130】上記ループ内のステップ S 511において、 $\text{dt}s$  には、フレームを 1 つデコードするたびに  $\text{Tfrm}$  ずつ加算されるが、これは、本実施形態が固定時間間隔  $\text{Tfrm}$  のフレーム発生を行う符号化方式を用いていることに由来する。ステップ S 512 では、CPU 412 は、今回の時間間隔  $\text{Ts}$  でデコードされるべきフレームの有無を判定している。ステップ S 512 の判定結果が否定、すなわち、もはや今回の時間間隔  $\text{Ts}$  でデコードされるフレームは無いと判定されると、CPU 412 は、上記のループ (ステップ S 509 ~ S 512) から抜け、ステップ S 513 に進む。ステップ S 513 では、CPU 412 は、変数  $\text{grid}$  を次の  $\text{grid}$  時刻に更新する。そして、関数を抜け、メインフロー (図 13) のステップ S 404 に判定結果 TRUE で RETURN する。

【0131】以上のアルゴリズムにより、図 9 で示したように、端末 102 内において、バッファ占有量  $\text{Sum}$  を常に  $\text{S\_target}$  の近傍でかつ  $\text{S\_target}$  を超えないように遷移させることができるとなる。従つて、複数機種の端末 102 があつて、バッファの総容量  $\text{Smax}$  が機種によって異なっていても、それぞれの端末 102 の  $\text{Smax}$  に応じて  $\text{S\_target}$  を適切な値に設定すれば、バッファのオーバーフローもアンダーフローも生じないようにすることができる。

【0132】なお、今回の例では、図 15 (A) のように 1 パケットに複数フレームを挿入するパターンでパケットを生成したが、代わりに、図 15 (B) のように、1 パケットに 1 フレームのフレームを挿入するパターンでパケットを生成することも可能である。この場合は、図 14 のステップ S 502 において、後半の不等式を  $\text{delta} + (\text{L} + \text{hdr}) <= \text{deltaMax}$  とし、ステップ S 504 の後半の等式を  $\text{delta} += (\text{L} + \text{hdr})$  とするだけでよい。

【0133】また、本実施形態では、説明を簡単にするために、固定時間間隔  $\text{Tfrm}$  でフレーム発生を行う符

号化方式を用いたが、使用する符号化方式一たとえばMPEG4ビデオ(ISO/IEC 14496-2)ーに合わせてデコード量算出アルゴリズムA2を設計すれば、必ずしも固定時間間隔のフレーム発生を行わなくて構わないことはいうまでもない。また、必ずしもフレーム単位でデータを扱うアルゴリズムでなくてもよく、例えばスライス単位、あるいはMPEG1やMPEG2システムストリームのパック単位でデータを扱うアルゴリズムであってもよい。

【0134】一方、図14のステップS502において、S\_targetの値が途中で変更されると、本アルゴリズムは瞬時に、変更された新しいS\_targetをターゲットとしてパケットの生成を行うようになる。S\_targetの値が途中で変更された場合のバッファ遷移の様子が、図10および図11に示されている。図10において、i=3の時刻にS\_targetがS\_target2に変更( $S_{target} < S_{target}2$ )されると、変更後しばらくは多量のフレームデータがパケット化され(図中、delta3やdelta4)、その結果、Sumが新しい目標値S\_target2の近傍に到達するようになる。

【0135】また、図11のように、i=2の時刻にS\_targetがS\_target3に変更( $S_{target}3 < S_{target}$ )されると、少量(delta4)または0(delta3)のフレームデータがパケット化される。その一方、デコードによりSumが消費されるので、やはりSumが新しい目標値S\_target3の近傍に到達するようになる。このような仕組みを利用すると、ネットワーク103の伝送能力(あるいは端末102の電波受信状態)に応じて、動的に端末102内のバッファ占有量Sumを増減させることが可能となり、以下に説明するような応用が可能となる。

【0136】図7(A)において、携帯電話701(図1の端末102と対応)を持ったユーザが、図中の矢印702ように、中継局B1の圏内から中継局B2の圏内へと移動する場合を考える。移動に伴い、携帯電話701からの呼を受け付ける業務が、中継局B1から中継局B2へと引き渡される。このとき、携帯電話701の受信電波強度は、おおむね図7(B)に示したグラフのように変化する。本モデルでは、説明を簡単にするために、電波強度が強から中(または中から強)に変わるところをネットワーク103の伝送能力に関する閾値A、中から弱(または弱から中)に変わるところを閾値B、弱から圏外(または圏外から弱)に変わるところを閾値Cとした。

【0137】図7(B)において、今、携帯電話701を持ったユーザが距離d1だけ移動し、伝送能力が閾値Aを下回ったとする。このとき携帯電話701は、図1

1に示されるように、S\_delayをより大きい値( $S_{target}2$ )に変更して、その値をサーバ101に通知する。これは、その後も進むと予想される伝送能力低下に備えて、サーバ101による新たなパケット生成および送出を促進させ、それにより、できるだけ長時間(これを $\Delta t$ とする)ぶんのデータを携帯電話701内のバッファに蓄積しておくためである。伝送能力が閾値Aを下回っても、閾値B以上である間は、まだパケットの転送ロスが発生する事がないので、このような伝送速度の高速化が可能である。

【0138】ユーザが移動して距離d2に達すると、伝送能力が閾値Bを下回って、パケットの転送ロスが発生し始める。このとき、携帯電話701は、図11に示されるように、S\_targetを小さい値( $S_{target}3$ )に変更して、その値をサーバ101に通知する。これは、その後もさらに進むと予想される伝送能力低下に備えて、できるだけサーバ101による新たなパケット生成および送出を抑制させるためである。パケット生成および送出を抑制するのは、次の理由による。

【0139】たとえば、携帯電話701が通信方式としてPHSのPIAFS方式を採用している場合、パケットの伝送ロスが発生すると、リンクレイヤであるPIAFS層のプロトコルに基づくデータ再送処理が行われる。再送処理中に、新たなパケット生成および送出が行われると、それが再送処理の邪魔をする結果となり、かえって好ましくないからである。

【0140】ユーザが移動して距離d3に達すると、伝送能力が閾値Cを下回って、一瞬、パケット転送が困難となる。しかし、ユーザがさらに移動して距離d4に達すると、伝送能力が閾値Bを上回り、かつ呼を受け付ける業務の引き渡し(ハンドオーバー)も完了しているので、携帯電話701は、今度はS\_target3を元の値S\_targetに戻して、その値をサーバ101に通知し、それによって、データの蓄積量(すなわちバッファ占有量Sum)を増加させる。なお、PHS等のハンドオーバー時間は、普通に人が歩く速さでもおおよそ2~3秒程度で完了するため、上記の $\Delta t$ をおおよそ3~4秒程度確保しておけば、ハンドオーバーが起こっても携帯電話701でのストリーミング再生を滞りなく継続することができる。

【0141】ところで、図11のように、ストリーム配信の途中でS\_targetの設定値がより小さな値に変更されると、図14のアルゴリズムにおいてステップS502の判定文がなかなか真にならず、次のフレームのデータを送出できないケースが起こりうる。このようなケースがたびたび発生すると、折角パケットを端末102に届けても、もはやそのパケット内のフレームデータを再生するべき時刻(Presentation Time)を経過してしまっており、データが無駄になってしまふことがある。このような場合は、再生時刻が経

過してしまったフレームデータをスキップした方が、無駄なデータをネットワーク103に流さないで済む分だけ効率的である。

【0142】図16は、図13のステップS404中の関数mkPacketの別の例を示すフローチャートである。図16の関数mkPacketには、サーバ101が送信速度を遅くする際に、再生時刻を過ぎたデータの送出をスキップするためのステップ(S601およびS602)が含まれている。すなわち、図16のアルゴリズムは、図14と比較して、ステップS601およびステップS602が追加されただけである。他のステップは、図14と全く同じであり、同一の参照番号が付されている。ステップS601では、CPU412は、今から送出しようとしているin番目のフレームデータが、0番目のフレームのデータでなく、かつ、端末102にて既にデコードされたとみなされるout番目のフレームデータより再生時刻が後かどうかを判定している。

【0143】この判定結果が真ならば、CPU412は、in番目のフレームのデータが端末での再生時刻に間に合うと見なし、ステップS503にてそのデータをパケット化し、端末102に送出する。偽の場合は、in番目のフレームデータを無かったものとみなし、ステップS602にてL=0とする。これにより、ステップS502では必ず真と判定され、かつステップS503のパケット化において、不要なフレームデータのコピーを行わずに、送出フレームを次に進めることができる。なお、このようなフレームスキップがあった場合は、デコーダ509での再生が時間Tfrmだけ飛ぶので、その旨を端末102に通知する情報が、図15(A), (B)のパケット中に記述されるものとする。例えば、ヘッダ内にそのような再生時刻情報を記述する領域を設ければよい。

【0144】図16に示したアルゴリズムは、MPEGオーディオのように各フレームどうしの優先順位(重要度)に差がない場合には、十分有効な手法である。しかし、MPEGビデオにおいては、従来例の紹介において既に説明したように、Iフレームであればそれ単独で意味のある画像を再構成することができるが、PやBのフレームでは、時間的に前後の参照フレームがなければ、意味のある画像を再構成することができない。この場合には、図16のアルゴリズムにおいてフレームの間引きを行う際に、再生時刻に間に合うIフレームを優先的に送出する一方、PやBのフレームを全てスキップすることで、ネットワーク103の転送速度が遅い状況においても、端末102に対してより高品位の映像を提供することが可能となる。

【0145】図17は、図13のステップS404中の関数mkPacketの、さらに別の例を示すフローチャートである。図17の関数mkPacketには、サ

ーバ101が送信速度を遅くする際に、優先度の低いデータと、優先度は高いが再生時刻を過ぎたデータとの送出をスキップするためのステップ(S505', S601, S602, S701およびS702)が含まれている。すなわち、図17のアルゴリズムは、図14と比較して、ステップS601, S602, S701およびS702が追加され、かつステップS505がステップS505'に置き換えられている。ステップS505'は、ステップS505に対し、関数Nextfrmに優先順位priの検出機能が加えられたものである。他のステップは、図14、図16と全く同じであり、同一の参照番号が付されている。

【0146】従って、図17のアルゴリズムは、図16と比較すると、ステップS701, S702が追加され、かつステップS505がS505'に置き換わっている。

【0147】図17のアルゴリズムを実行するには、端末102が検出した受信状態を示す情報(受信状態情報)を、端末102からサーバ101に通知する機能が必要となる。このような機能を持ったサーバ・クライアント・システムの構成例を、図18に示す。図18において、端末102は、受信状態を検出する検出部801を備えている。端末102とサーバ101との間には、検出された受信状態情報を端末102からサーバ101に通知する通知部802が設けられる。サーバ101は、保持部803を備えており、通知された受信状態情報を保持する。

【0148】再び図17において、mkPacket関数が呼び出されると、ステップS501に先立って、ステップS701が実行される。ステップS701において、サーバ101(のCPU412)は、保持部803内の情報(端末102側の受信状態)を参照して、ネットワーク103の伝送能力が閾値Bを下回るか否かを判定する。この判定の結果、閾値Bを下回っていればslowflagを真とし、そうでなければ偽とする。

【0149】ステップS505'では、次のフレームの優先度が検出され、続くステップS702では、そのフレームのデータの優先度が高く、かつslowflagが真であるか否かが判定される。この判定結果が肯定、すなわちネットワーク103の転送速度が遅いことを示すslowflagが真であり、かつ優先度の高いフレームである場合、ステップS601に進んで、再生時刻が経過してしまったフレームか否かが判定される。一方、判定結果が否定である場合、ステップS602に進んで、L=0とされる(つまり、たとえ再生時刻に間に合うようであっても、そのフレームはスキップされる)。後の処理は、図14や図16の処理と全く同様である。

【0150】以上のように、本実施形態によれば、端末102が、自身のバッファ容量とネットワーク103の

伝送能力とに応じた目標量を決定し、さらに、目標量を伝送能力で除して得られる値を超えない範囲内で、遅延時間を決定する。サーバ101は、こうして端末102が決定した目標量および遅延時間に基づいて送信速度を制御するので、端末102のバッファ容量が機種によって異なっていても、ネットワーク103の伝送能力が変動しても、バッファ量および伝送能力に応じた送信速度制御が行え、その結果、バッファのアンダーフローやオーバーフローによるストリーミング再生の破綻を回避することが可能となる。しかも、目標量とは独立に遅延時間が決定されるので、ストリーミング再生の破綻回避と、頭出し時の待ち時間短縮とを互いに両立させることができる。

#### 【図面の簡単な説明】

【図1】本発明の一実施形態に係るストリーミング方法を実行するサーバ・クライアント・システムの構成例を示すブロック図である。

【図2】図1のサーバ101の構成を示すブロック図である。

【図3】図1の端末102の構成を示すブロック図である。

【図4】図1のシステムの全体動作を説明するためのシーケンス図である。

【図5】図1の端末102の動作を示すフローチャートである。

【図6】図3のROM502の記憶内容を示す図である。

【図7】あるエリアにおける電界強度の分布（A）と、端末の移動に伴う伝送能力の変化（B）とを示す模式図である。

【図8】図5のステップS107の詳細を示すフローチャートである。

【図9】図1のサーバ101が行う送信速度制御によって、端末102のバッファ占有量がどのように遷移するか（S\_targetに近づいていく様子）を示す図である。

【図10】バッファ占有量がS\_targetの近傍で遷移している状態で、S\_targetの値がより大きな値（S\_target2）に変更された場合に、図1のサーバ101が行う送信速度制御によって、端末102のバッファ占有量がどのように遷移するかを示す図である。

【図11】バッファ占有量がS\_targetの近傍で遷移している状態で、S\_targetの値がより小さな値（S\_target3）に変更された場合に、図1のサーバ101が行う送信速度制御によって、端末102のバッファ占有量がどのように遷移するかを示す図である。

【図12】図1のサーバ101が行う送信速度制御アルゴリズムの一例を示すフローチャートである。

【図13】図9～図11のバッファ遷移を実現するためにサーバ101によって行われる送信速度制御アルゴリズムの別の例を示したフローチャートである。

【図14】図13のステップS404中の関数mkPacketの一例を示すフローチャートである。

【図15】図1のサーバ101が生成するパケットの構成例（Aは1パケットに複数フレームを挿入する場合、Bは1パケットに1フレームを挿入する場合）を示す図である。

【図16】図13のステップS404中の関数mkPacketの別の例を示すフローチャートである。

【図17】図13のステップS404中の関数mkPacketの、さらに別の例を示すフローチャートである。

【図18】本発明の一実施形態に係るストリーミング方法を実行するサーバ・クライアント・システムの別の構成例を示すブロック図である。

【図19】従来のストリーミング方法を説明するための図である（Aはビデオフレーム、Bはバッファ占有量の遷移、Cは従来端末の構成例）。

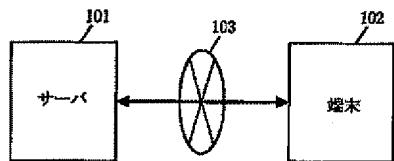
【図20】従来のストリーミング方法を実行するサーバ・クライアント・システムの構成例を示すブロック図である。

【図21】受信バッファを追加することによってバッファ占有量の遷移がどのように変化するかを説明するための図である（Aが追加前、Bが追加後）。

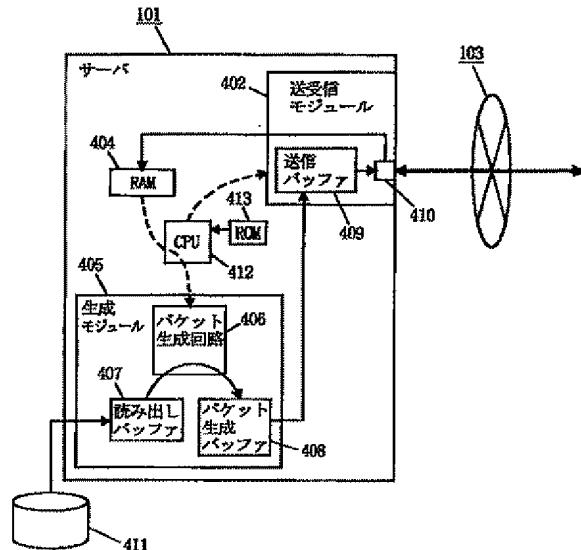
#### 【符号の説明】

- 101 サーバ
- 102 端末
- 103 ネットワーク
- 402, 507 送受信モジュール
- 404 RAM
- 405 生成モジュール
- 406 パケット生成回路
- 407 読み出しバッファ
- 408 パケット生成バッファ
- 409 送信バッファ
- 410, 506 ネットワークコントローラ
- 411 喧音デバイス
- 412, 503 CPU
- 413, 502 ROM
- 505 受信バッファ
- 508 デコーダバッファ
- 509 デコーダ
- 510 再生モジュール
- 511 表示デバイス

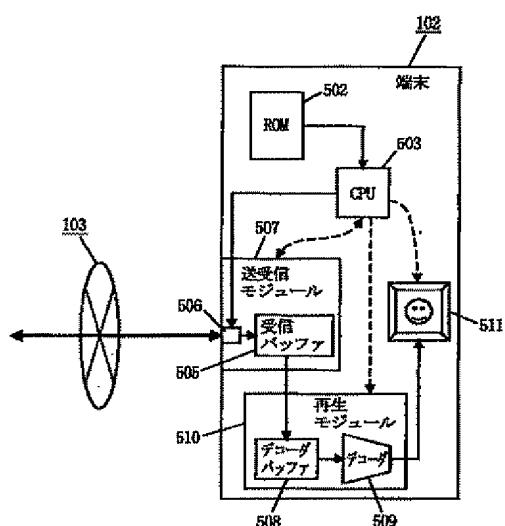
【図1】



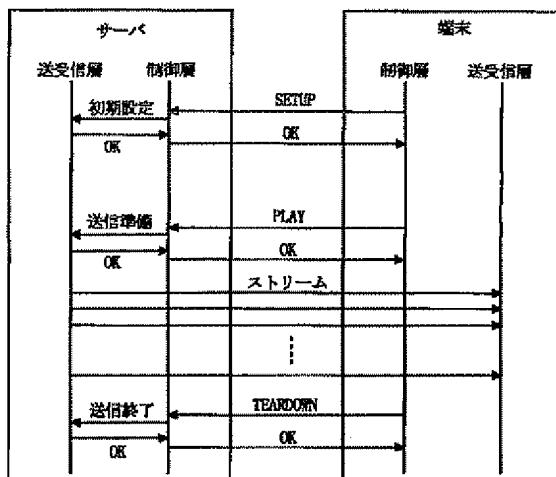
【図2】



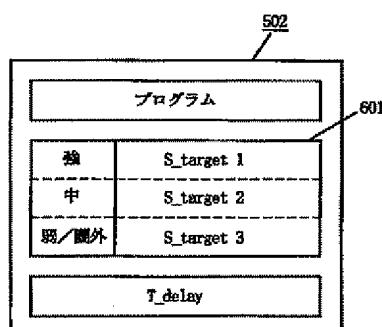
【図3】



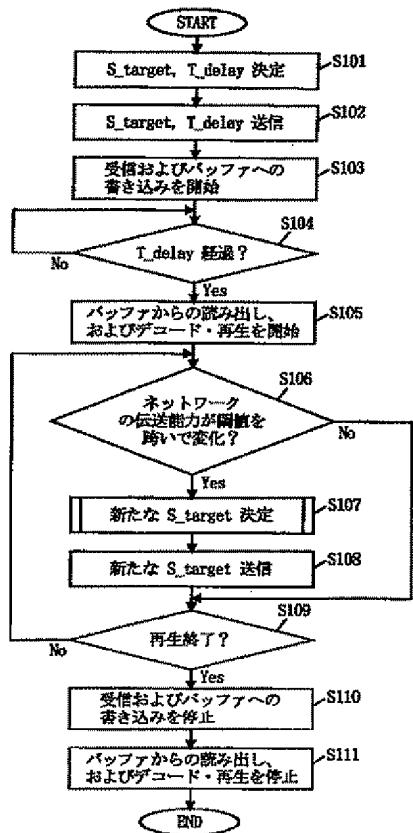
【図4】



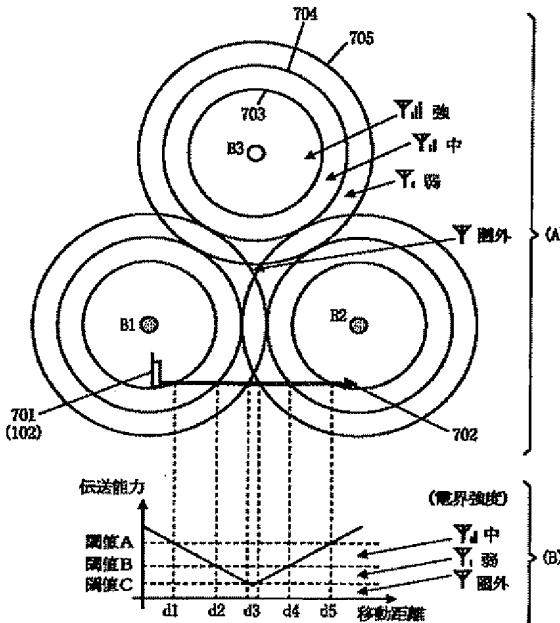
【図6】



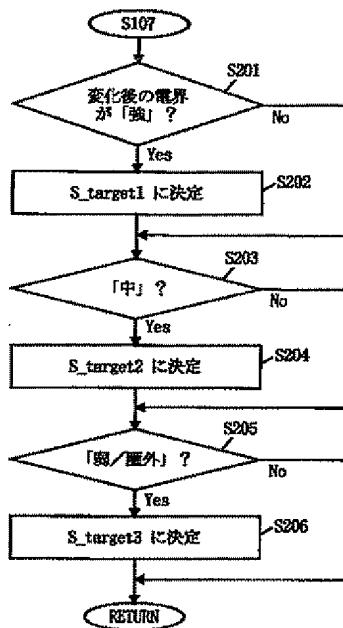
【図5】



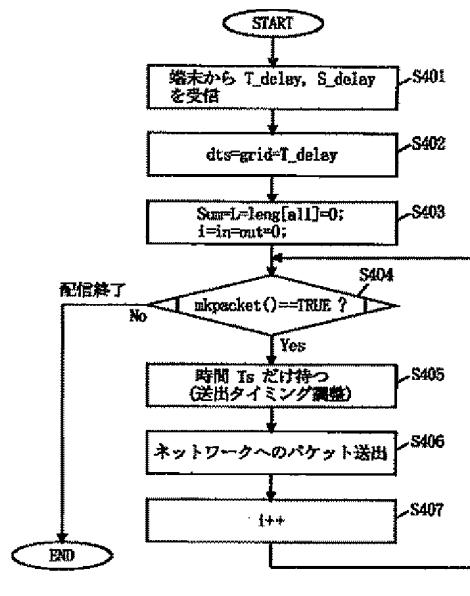
【図7】



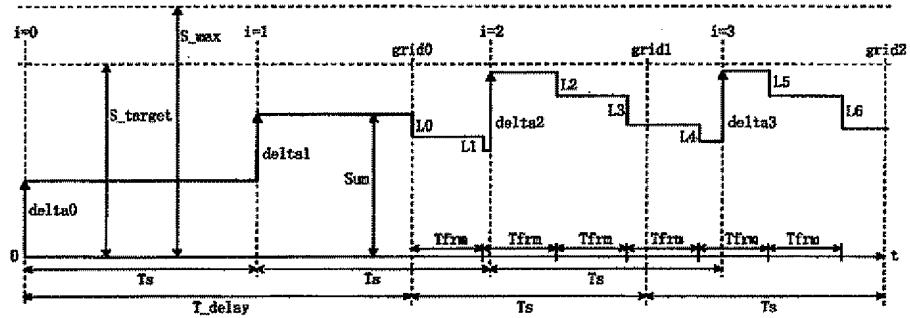
【図8】



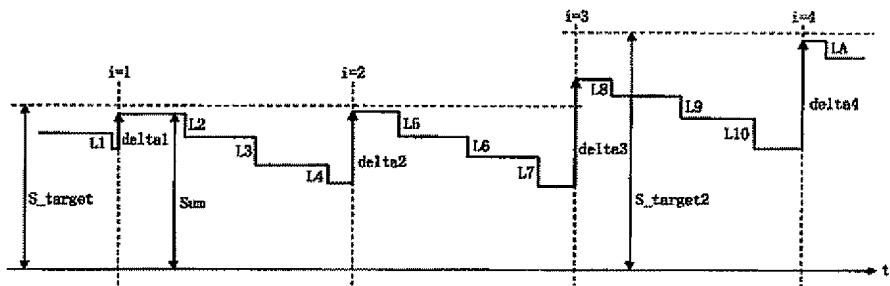
【図13】



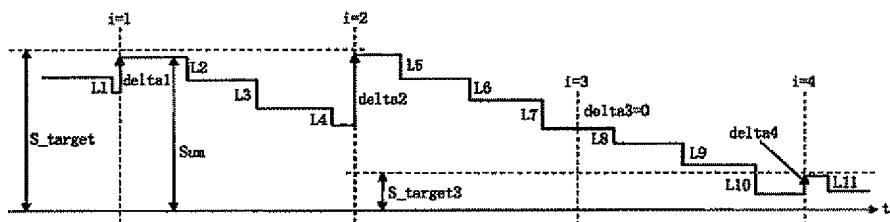
【図9】



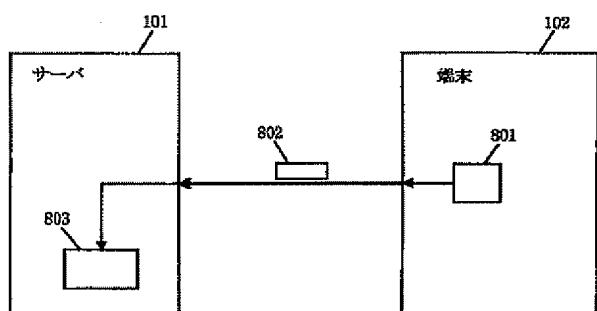
【図10】



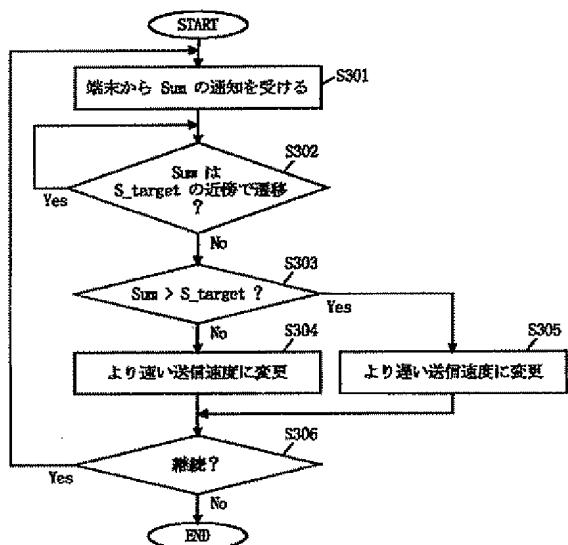
【図11】



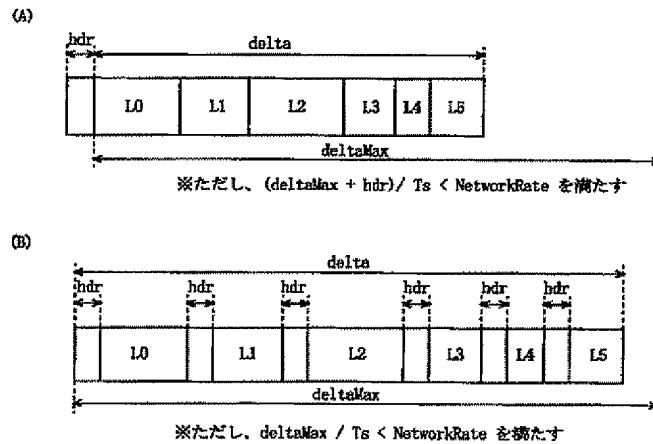
【図18】



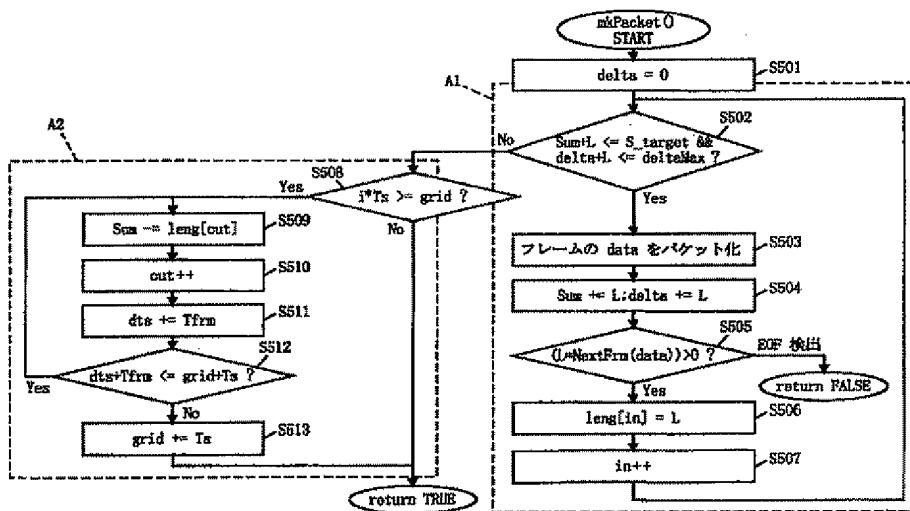
【図12】



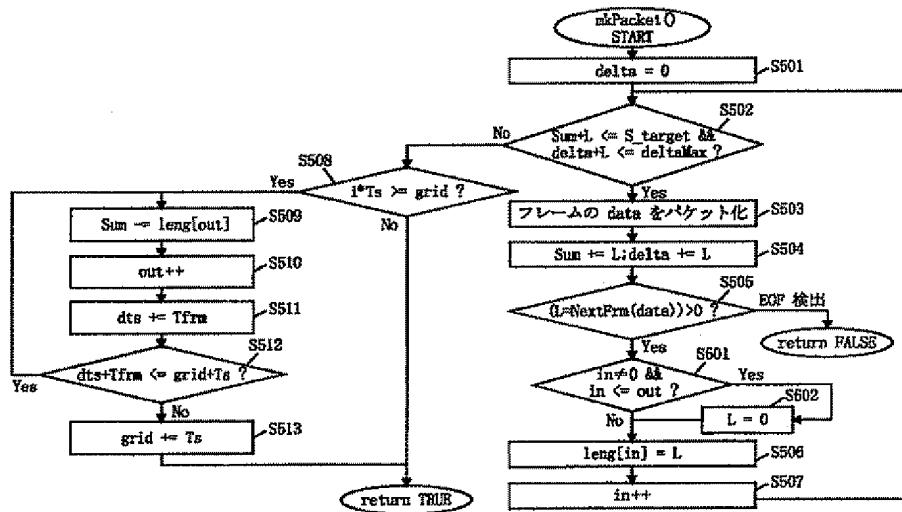
【図15】



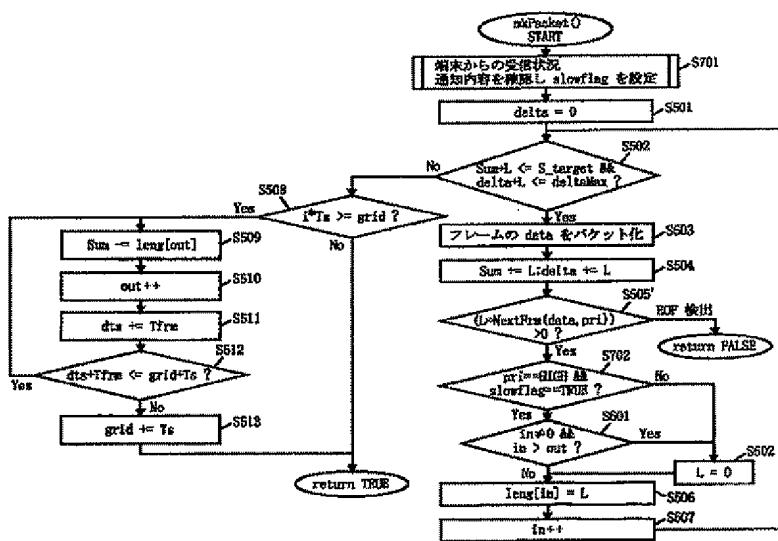
【図14】



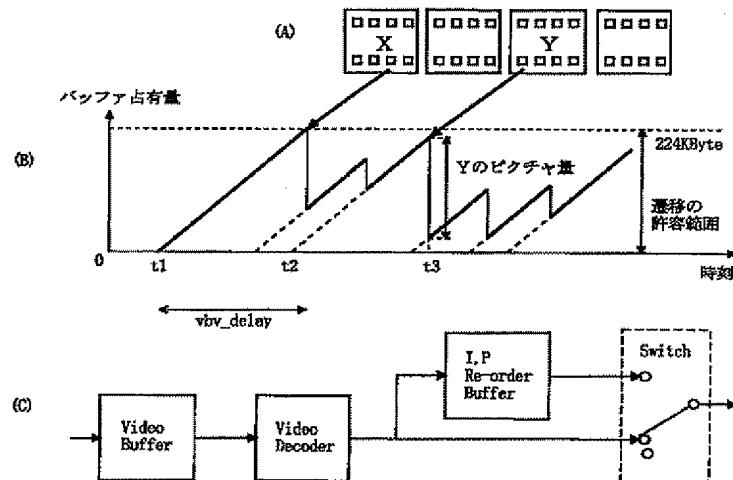
### 【図16】



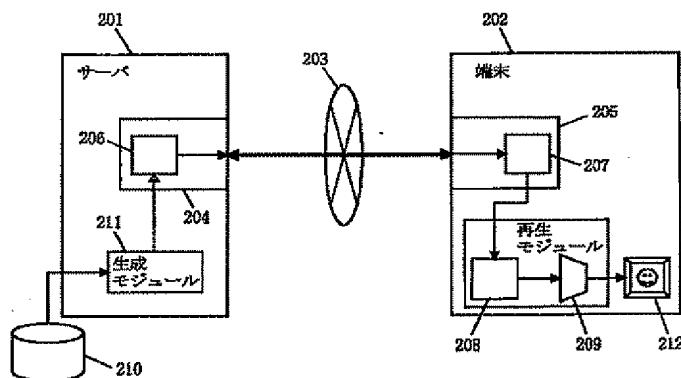
〔圖 17〕



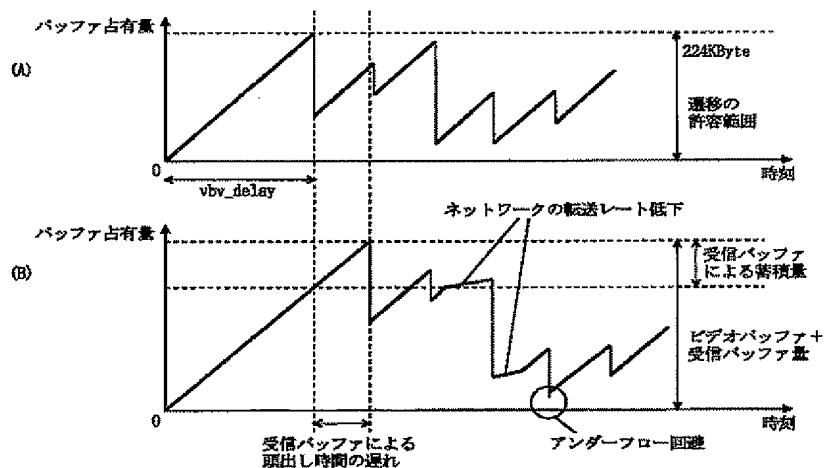
【図19】



【図20】



【図21】



## フロントページの続き

(72) 発明者 藤田 隆久  
大阪府門真市大字門真1006番地 松下電器  
産業株式会社内

F ターム(参考) 5C059 KK35 RB02 RB04 RE16 RE20  
SS08 SS10 SS20 TA71 TC16  
TO38 TD14 UA32 UA39  
5K030 HA08 HB02 HB21 JT10 KA01  
KA03 KA07 LC01 LE16 LE17  
MB15  
5K034 AA05 CC02 CC05 DD02 EE10  
HH42 MM08

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-084339  
 (43)Date of publication of application : 22.03.2002

(51)Int.Cl. H04L 29/08  
 H04L 12/56  
 H04L 13/08  
 H04N 7/24

(21)Application number : 2001-202147 (71)Applicant : MATSUSHITA ELECTRIC IND CO LTD  
 (22)Date of filing : 03.07.2001 (72)Inventor : HARUMOTO HIDEAKI  
 HORIUCHI YUUKI  
 FUJITA TAKAHISA

(30)Priority

Priority number : 2000204632 Priority date : 06.07.2000 Priority country : JP

## (54) STREAMING METHOD AND SYSTEM FOR EXECUTING THE SAME METHOD

### (57)Abstract:

**PROBLEM TO BE SOLVED:** To provide a streaming method, capable of preventing failure of streaming reproduction, even if the buffer capacities of terminals are different according to the kinds of devices, and even if the transmitting capabilities of a network fluctuate, and to make both prevention of the failure of the streaming reproduction and shortening of waiting time compatible at program searching.

**SOLUTION:** A terminal decides a target quantity ( $S_{target}$ ) of a stream to be stored in own buffer concerning the capacity of own buffer and the transmitting capabilities of a network; and it decides a delay time ( $T_{delay}$ ), and writes the leading data of the stream in own buffer, until the data are read and started to be reproduced arbitrarily within a range, which is not beyond the value obtained by removing the buffer capacity by the transmitting capabilities, and notifies a server of the target quantity and the delay time. The server controls the transmission speed, so that the transition of the buffer occupancy quantity (Sum) of the terminal can be realized without exceeding the target quantity within the target quantity, based on the notification.

